



UNIVERSIDADE SALVADOR – UNIFACS
PROGRAMA DE PÓS-GRADUAÇÃO
MESTRADO PROFISSIONAL EM SISTEMAS E COMPUTAÇÃO

ANTONIO CARLOS CABRAL VILAR

UM AMBIENTE DE SIMULAÇÃO PARA APLICAÇÕES
DINÂMICAS DE TV DIGITAL COM TRANSMISSÃO AO VIVO

Salvador
2010

ANTONIO CARLOS CABRAL VILAR

**UM AMBIENTE DE SIMULAÇÃO PARA APLICAÇÕES
DINÂMICAS DE TV DIGITAL COM TRANSMISSÃO AO
VIVO**

Dissertação apresentada ao Curso de Mestrado Profissional em Sistemas e Computação, Universidade Salvador - UNIFACS, como requisito parcial para obtenção do grau de Mestre.

Orientador: Prof. Celso Alberto Saibel Santos

Salvador
2010

Ficha Catalográfica

(Elaborada pelo Sistema de Bibliotecas da Universidade Salvador - UNIFACS)

Vilar, Antonio Carlos Cabral

Um ambiente de simulação para aplicações dinâmicas de TV digital com transmissão ao vivo. / Antonio Carlos Cabral Vilar. – Salvador, 2010.

79 p. : il.

Dissertação apresentada ao Curso de Mestrado em Sistemas e Computação da Universidade Salvador – UNIFACS, como requisito parcial para a obtenção do grau de Mestre.

Orientador : Prof. Dr. Celso Alberto Saibel Santos.

1. TV Digital. 2. Interatividade. I. Santos, Celso Alberto Saibel, orient. II. Universidade Salvador – Unifacs. III. Título.

CDD: 004

TERMO DE APROVAÇÃO

ANTONIO CARLOS CABRAL VILAR

UM AMBIENTE DE SIMULAÇÃO PARA APLICAÇÕES
DINÂMICAS DE TV DIGITAL COM TRANSMISSÃO AO
VIVO

Dissertação aprovada como requisito parcial para obtenção do grau de Mestre Sistemas e Computação, Universidade Salvador - UNIFACS, pela seguinte banca examinadora:

Celso Alberto Saibel Santos – Orientador _____
Doutor em Informatique Fondamentale et Parallelisme pela Université Paul Sabatier de
Toulouse III
Universidade Federal da Bahia - UFBA

Cassio Vinicius Serafim Prazeres _____
Doutor em Computação pela Universidade de São Paulo - USP
Universidade de São Paulo - USP

Cesar Teixeira _____
Doutor em Computação pela Universidade Federal de São Carlos - UFSC
Universidade Federal de São Carlos - UFSC

Salvador, 14 de junho de 2010

AGRADECIMENTOS

A minha esposa Juliana, por todo seu amor e alegria. Princesa, te amo muito. Sem você não conseguiria.

A meu filho Alexandre, que nasceu no mesmo ano em que ingressei no mestrado. Filho, não existe maior bálsamo para qualquer adversidade no mundo, que o seu sorriso. Te amo muito filhão.

A meus pais Antonio e Marta, por todo amor, carinho e compreensão durante toda minha vida. Amo vocês.

A meus irmãos, Rodrigo e Mariana, parceiros de conquistas durante toda minha vida. Se cuidem, amo vocês.

Aos meus sogros Vani e Lígia, por todo amor e apoio prestado durante nossa convivência.

Ao meu cunhado Paulo por todo amor, suporte e presteza prestados a mim e a minha família.

Aos meus tios, tias, primos e primas. Em especial, tio Luis, que apoiou e acreditou neste empreendimento.

A todos amigos, pelos momentos maravilhosos de diversão e pelo ombro amigo na adversidade.

A todos professores, queridos colegas, que participam de forma tão ativa na minha formação. Em especial, meu grande amigo, prof. Euclerio e meu orientador prof. Celso.

Obrigado a todos! De coração.

“Hoje é o dia de sua vitória sobre quem você foi ontem;
Amanhã é a sua vitória lutando contra menos gente”

Miyamoto Musashi

RESUMO

A TV Digital interativa traz a possibilidade de executar programas de computador (aplicações interativas) no aparelho receptor de TV. Tal característica torna possível assistir TV com recursos que permitem a interação do telespectador (usuário) com os programas, apresentando um novo paradigma ao ambiente de TV. Surgem então, para o ambiente de TV, uma série de desafios com relação à implementação, distribuição, direitos autorais, usabilidade, suporte da infra-estrutura, etc. Este trabalho apresenta um ambiente de simulação para aplicações dinâmicas de TV Digital, com transmissão ao vivo, e apresenta os desafios e as soluções encontradas durante o desenvolvimento de uma aplicação dinâmica de TV, capaz de interagir com o conteúdo transmitido ao vivo. O ambiente desenvolvido implementa funcionalidades típicas de um ambiente real de TV Digital, fornecendo uma visão simulada da experiência do usuário, do operador de TV e do desenvolvedor da aplicação interativa.

Palavras-chave: TV Digital. Interatividade. Eventos. Simulador. Ao vivo.

ABSTRACT

Interactive Digital TV makes possible to execute computer programs (interactive applications) in a digital TV receiver. This feature allows gives the user capacity to interact with the TV show while watching TV, which presents a new set of challenges for the TV environment, like distribution, usability, rights management, infrastructure support, development, etc. This work presents a simulation environment for dynamic applications for digital TV, with live transmission, and presents the challenges and solutions found during the development of the environment as the development of a dynamic digital TV application, capable to interact with a live content TV show. The developed environment implements typical functionalities found in real Digital TV environment, giving a simulated view of the user, developers and TV operators experience.

Keywords: Digital TV. Interactive. Events. Simulator. Live.

LISTA DE FIGURAS

Figura 1 - Aplicações Interativas de TVD	14
Figura 2 - Padrões Internacionais de TVD.....	20
Figura 3 - Transmissão Hierárquica e Recepção Parcial ISDB.....	21
Figura 4 - Transmissão de vídeo, áudio e dados extras	25
Figura 5 - Arquitetura de camadas de um sistema computacional de TVD.....	26
Figura 6 - Ginga-Core.....	28
Figura 7 - Ginga-NCL.....	29
Figura 8 - Arquitetura Ginga-J	30
Figura 9 - Serviço público jurídico, Reino Unido	31
Figura 10 - Seleção interativa de áudio.....	31
Figura 11 - (A) Visão posterior do SBT Zinwell ZBT-620A. No detalhe, conector USB. (B) Seleção do dispositivo USB para acesso a dados no armazenamento externo	33
Figura 12 - Carrossel de Dados	34
Figura 13 - Aplicação interativa escrita em NCL.....	37
Figura 14 - Um modelo de aplicação interativa para TV em uma abordagem orientada a objetos	46
Figura 15 - Modelo de dados extras difundido em uma aplicação de jogo de futebol (BEC). 48	
Figura 16 - Modelo de dados extras no receptor (BEC) em uma aplicação de jogo de futebol	49
Figura 17 - Estrutura do ambiente de simulação	52
Figura 18 - Diagrama de Classes do InteractiveSimSrv	54
Figura 19 - Diagrama de classes do InteractiveSimClient	55
Figura 20 - InteractiveSimClient	56
Figura 21 - Interface para definição de dados estruturados	57
Figura 22 - Interface para geração e anotação de eventos	58
Figura 23 - Dinâmica de funcionamento para aplicações de conteúdo e formato dinâmicos - <i>Estruturação e Distribuição</i>	59
Figura 24 - Dinâmica de funcionamento para aplicações de conteúdo e formato dinâmico - <i>Operação</i>	60
Figura 25 - Estrutura de diretórios para dados dinâmicos e estáticos.....	62
Figura 26 - Raiz de diretório: Arquivos em XML representando a ocorrência de eventos	63
Figura 27 - Arquivo XML representando a ocorrência de um gol	64
Figura 28 - Seleção, no receptor, de informação de acordo com o perfil do usuário	64
Figura 29 - Comportamento gerado de acordo com o perfil do usuário.....	65
Figura 30 - Apresentação de mídias dinâmicas (extras) de acordo com solicitação do usuário	66
Figura 31 - Envio de dados pelo carrossel e tratamento de arquivo corrompido.....	68

LISTA DE TABELAS

Tabela 1 - Percentual de bens duráveis por domicílios separados por região.....	11
---	----

SUMÁRIO

CAPÍTULO 1	11
1 INTRODUÇÃO	11
1.1 MOTIVAÇÃO	13
1.2 OBJETIVO.....	15
1.3 CONTRIBUIÇÃO.....	15
1.4 ESTRUTURA DO TRABALHO.....	16
CAPÍTULO 2	17
2 TECNOLOGIA DE TV DIGITAL	17
2.1 PADRÕES INTERNACIONAIS DE TV DIGITAL	17
2.2 ISDTV – O PADRÃO BRASILEIRO	21
2.2.1 As Escolhas do SBTVD	22
2.3 CONSIDERAÇÕES FINAIS	23
CAPÍTULO 3	25
3 APLICAÇÕES INTERATIVAS PARA TV DIGITAL	25
3.1 MIDDLEWARES	26
3.2 O GINGA	27
3.2.1 Arquitetura	27
3.3 INTERATIVIDADE	30
3.4 APLICAÇÕES INTERATIVAS.....	32
3.5 O CARROSSEL DE DADOS.....	33
3.6 DESENVOLVIMENTO DE APLICAÇÕES INTERATIVAS: CONCEITOS.....	34
3.6.1 Paradigma Declarativo	35
3.6.1.1 A Parte Declarativa do Ginga: Ginga-NCL	36
3.6.1.2 A Linguagem de <i>Script</i> Lua	37
3.6.2 Paradigma Procedural	38
3.6.2.1 A Parte Procedural do Ginga: Ginga-J.....	39
3.7 CONSIDERAÇÕES FINAIS	39
CAPÍTULO 4	41
4 PROGRAMAS INTERATIVOS DE TV	41
4.1 MODELOS PARA EDIÇÃO DE APLICAÇÕES AO VIVO	42
4.1.1 Modelo baseado em eventos	44
4.2 ESTRUTURAÇÃO DE CONTEÚDO INTERATIVO	45

4.3	MODELAGEM DE UM PROGRAMA INTERATIVO DE TV AO VIVO: UMA PARTIDA DE FUTEBOL.....	47
4.4	CONSIDERAÇÕES FINAIS	49
	CAPÍTULO 5.....	51
5	AMBIENTE DE SIMULAÇÃO DE PROGRAMAS INTERATIVOS	51
5.1	O AMBIENTE DE SIMULAÇÃO <i>INTERACTIVESIM</i>	51
5.2	VISÃO GERAL DA ESTRUTURA DO AMBIENTE.....	52
5.3	ARQUITETURA DO AMBIENTE	54
5.3.1	Dinâmica de Funcionamento.....	58
5.4	CONSIDERAÇÕES FINAIS	60
	CAPÍTULO 6.....	61
6	ESTUDO DE CASO.....	61
6.1	APLICAÇÃO INTERATIVA AO VIVO – JOGO DE FUTEBOL.....	61
6.1.1	Geração, Transmissão e Tratamento de Eventos	63
6.2	CONSIDERAÇÕES.....	66
	CAPÍTULO 7.....	70
7	CONCLUSÕES	70
7.1	CONTRIBUIÇÕES.....	73
7.2	TRABALHOS FUTUROS.....	74
	REFERÊNCIAS	76

CAPÍTULO 1

1 INTRODUÇÃO

A digitalização da transmissão dos sinais de televisão no Brasil é vista como uma ação estratégica para o desenvolvimento do País (BRASIL, 2003), na medida em que sua presença em aproximadamente 90% dos lares brasileiros (Tabela 1) faz da televisão o principal instrumento de informação e entretenimento da população, atingindo 98% dos brasileiros, segundo dados do Instituto Brasileiro de Geografia e Estatística (IBGE, 2009). Economicamente, a transição do sistema analógico para o digital relaciona-se a uma série de fatores mercadológicos, tais como o fortalecimento da indústria, a formação de pessoal especializado e a criação de novas oportunidades para o surgimento de serviços. Socialmente, representa o surgimento de novas oportunidades de trabalho, a possibilidade de estender a inclusão digital a toda população e o acesso a uma rede universal de ensino a distância com significativo impacto educacional e cultural. Fortalece ainda a pesquisa, o ensino e o desenvolvimento tecnológico no País, isso tudo, sem mencionar benefícios como a melhoria da qualidade do áudio e do vídeo transmitidos, o uso mais eficiente do espectro de transmissão, menor custo de transmissão e consumo de energia, capacidade de interação (*Interactive Television - iTV*), transmissão de dados, flexibilidade, etc.

Tabela 1 - Percentual de bens duráveis por domicílios separados por região

Grandes Regiões	Percentual de domicílios com alguns bens duráveis e serviços de acesso à comunicação no total de domicílios particulares permanentes (%)											
	Iluminação elétrica	Telefone		Fogão	Filtro de água	Geladeira	Freezer	Máquina de lavar roupa	Rádio	Televisão	Microcomputador	
		Total	Somente móvel celular								Total	Com acesso à Internet
Brasil	98,6	82,1	37,6	98,2	51,6	92,1	16,0	41,5	88,9	95,1	31,2	23,8
Norte	94,9	72,4	49,0	96,7	31,4	83,9	15,3	26,7	76,4	90,0	17,4	10,6
Nordeste	97,0	66,8	43,9	96,1	49,1	81,5	7,0	15,5	82,4	91,7	15,7	11,6
Sudeste	99,8	88,9	29,3	99,4	65,0	97,3	15,6	54,3	93,0	97,6	40,0	31,5
Sul	99,4	89,8	40,8	99,0	22,6	97,2	31,5	59,0	94,9	96,4	38,5	28,6
Centro-Oeste	99,2	87,9	47,7	98,8	60,4	95,1	18,6	34,4	86,4	94,6	30,9	23,5

Fonte: IBGE (2009)

Em 26 de Novembro de 2003, o decreto presidencial 4.901 instituiu o Sistema Brasileiro de TV Digital Terrestre, o SBTVD. O decreto 4.901 teve, entre outros objetivos, a definição de todo o processo de digitalização do atual sistema televisivo brasileiro. Atualmente, algumas cidades brasileiras já iniciaram as transmissões digitais. Apesar disso, apenas uma pequena parte das funcionalidades esperadas da TV Digital já estão disponíveis aos usuários. Esse fato se deve, sobretudo, ao processo lento de adoção da nova tecnologia que envolve o desenvolvimento de um *middleware*. O *middleware* é uma camada de *software* responsável por manter a compatibilidade entre os sistemas computacionais (desenvolvidos por fabricantes de televisores, conversores e *set-top-boxes*) e as aplicações interativas. No Brasil, esta camada de *software* foi batizada de Ginga e tem como objetivos o suporte à interação dos usuários com o conteúdo televisivo e a possibilidade de incorporação de funcionalidades e serviços típicos de computadores ao televisor fazendo do Ginga, um poderoso instrumento de inclusão digital e entretenimento para a população brasileira.

De acordo com Tavares (2007) a TV digital traz consigo um novo modelo de serviço que torna possível a execução de aplicações interativas transmitidas juntas ao conteúdo principal de áudio e vídeo. Neste novo modelo um canal de retorno, denominado canal de interatividade, pode ser utilizado para estabelecer a ligação dos aparelhos receptores com as operadoras de TV, provedores de serviços, redes e até mesmo, outros usuários. Como consequência, uma emissora de TV passa a assumir uma postura mais ampla, comportando-se como uma produtora de conteúdo multimídia e de serviços, passando também a enviar e receber periodicamente dados dos “agora usuários” (antigos telespectadores) dos serviços associados aos programas interativos que estão sendo difundidos. A aplicação deste novo modelo de serviço diferencia então a TV digital, que envolve os sistemas de codificação, transmissão e decodificação de conteúdo de áudio e vídeo, da TV digital Interativa, que adiciona à TV digital a possibilidade do usuário participar de forma ativa da programação.

Porém, dentro desta perspectiva, a emissora de TV continua como elemento centralizador na produção de conteúdo, o que dificulta a entrada de novos participantes em sua cadeia produtiva. Uma forma de contornar este problema seria a adoção de um modelo interativo baseado em eventos, que tanto poderiam ser gerados pelas emissoras, quanto por fontes externas, via canal de interatividade (ou canal de retorno). Neste modelo uma aplicação sendo executada no receptor seria responsável pelo tratamento dos eventos recebidos e, a partir destes, poderia montar de forma dinâmica o ambiente com o qual o usuário interage. Tal

modelo permite que participantes externos a cadeia de produção das emissoras possam criar aplicações capazes de interagir com o conteúdo transmitido, aumentando assim o potencial de penetração destas aplicações na medida em que esses novos integrantes da cadeia produtiva trazem o aumento da oferta de aplicações voltadas para nichos, ou perfis específicos.

1.1 MOTIVAÇÃO

Aplicações interativas para TV digital mostram-se como um meio promissor para modificar a maneira como se vê e se faz TV. Tal potencial ganha forma, na medida em que decodificadores digitais adquirem recursos e funcionalidades que permitem a execução de aplicativos. Em teoria, estes aplicativos poderiam ser providos tanto pela emissora de TV, à qual o decodificador encontra-se sintonizado, enviados juntos ao fluxo de vídeo e áudio, como a partir de fontes de armazenamento locais (como um *pen-drive*) e remotas, como através da *Web* (MARQUES, 2008).

Desta forma, estes aplicativos, quando solicitados, podem fornecer conteúdo de natureza variada ao telespectador, propiciando o ingresso de uma série de novos participantes na cadeia de produção de conteúdo audiovisual para TV, inclusive o próprio usuário (PIMENTEL, 2010). Neste sentido, a TV Digital tem a possibilidade de apresentar ao telespectador um conteúdo enriquecido, que tanto contribui para a imersão do usuário na programação, como também possibilita o fornecimento de uma série de serviços e facilidades. Para ilustrar o cenário, podem-se citar aplicações de correio eletrônico, mensagens instantâneas, compras *on-line*, jogos, exibição de informações adicionais sobre paisagens, produtos ou atores de um programa e previsão climática, dentre outras (Figura 1). Estas aplicações, comuns no ambiente Web, embora se mostrem viáveis ao ambiente de interatividade de TV Digital, em alguns casos ainda requerem estudos mais significativos no que diz respeito a questões como segurança, usabilidade, direitos autorais, etc.



Figura 1 - Aplicações Interativas de TVD

Sob esta mesma perspectiva, também há a possibilidade de em transmissões ao vivo, como um jogo de futebol, adicionar conteúdo extra ao fluxo de vídeo e áudio, com informações a respeito do andamento da partida, tal como lances perigosos, faltas e gols. Estas informações, quando apresentadas em um formato padronizado, podem possibilitar ao usuário acesso a conteúdos, oriundos de outras fontes que não a emissora que controla o canal de transmissão TV. A idéia é contrária ao modelo vigente, com produção e distribuição centralizado pelas emissoras. Entretanto, se o decodificador for encarado como uma plataforma computacional convergente, dois cenários podem ser visualizados:

- a) O conteúdo apresentado na TV pode ser gerado a partir da integração de conteúdos de diferentes fontes, de forma semelhante ao que ocorre na Web (MARQUES, 2008, 2009a).
- b) Diferentes dispositivos podem ser integrados ao decodificador para gerar ou consumir o conteúdo da TV (CESAR, 2009).

Outro aspecto importante para motivar o desenvolvimento deste trabalho se deve ao seguinte fato: Com o aumento da banda para usuários finais, aplicações de internet passaram a habitar a rotina de entretenimento dos usuários de TV que, por meio da rede, podem agir de forma ativa sob seu conteúdo. Essa migração para o PC, de parte dos usuários que antes costumava se entreter com a TV, acarreta em perda de audiência para as emissoras de TV, reduzindo assim sua receita. A capacidade de executar aplicações interativas personalizadas oferece as emissoras uma chance de recuperar parte de sua audiência que migrou para o PC.

1.2 OBJETIVO

O objetivo deste trabalho é propor um modelo para a concepção de programas ao vivo baseado (i) na estruturação do conteúdo extras ao fluxo principal de áudio e vídeo, (ii) na especificação dos tipos de eventos gerados durante a transmissão do programa e (iii) no acesso ao conteúdo estruturado e no tratamento dos eventos pelos decodificadores.

A validação do modelo está baseada no estudo de caso de uma aplicação desenvolvida para simular o ambiente televisivo digital, tanto na parte responsável pela geração de conteúdo, o fluxo principal de áudio e vídeo, e dados extras, quanto no processamento e na apresentação dos mesmos.

1.3 CONTRIBUIÇÃO

O Sistema Brasileiro de TV Digital já se encontra em funcionamento em algumas cidades brasileiras (FORUM, 2010), transmitindo conteúdo digital em diferentes resoluções, e a venda de *set-top-boxes*, compatíveis com o sistema, já é uma realidade na rede de varejo de eletroeletrônicos. No entanto, a apresentação e desenvolvimento de aplicações interativas ainda são incipientes, se restringindo ao ambiente acadêmico, testes realizados por redes de TV e algumas empresas de tecnologia com interesse no desenvolvimento de programas interativos de TV. Esses interessados, embora tenham acesso a *set-top-boxes* embarcados com versões do Ginga, ainda sofrem com a imaturidade de alguns sub-componentes do sistema, sobretudo no que diz respeito ao sub-sistema procedural (Ginga-J), responsável pelas aplicações escritas em Java, e sobre a falta de definição com relação a estruturação dos dados extras, multiplexados ao fluxo principal de áudio e vídeo, moderado pelas emissoras de TV.

Diante deste cenário, a apresentação de um ambiente para prototipagem para aplicações interativas e o validação de um modelo de estruturação de dados extras, capaz de possibilitar o tratamento de eventos dinâmicos gerados pelas emissoras, contribui para elaboração de um padrão de serviço televisivo onde se possibilita que aplicações interativas apresentem um comportamento dinâmico gerado em função de eventos processados, enviados pela emissora de TV. O tratamento a estes eventos pode ser realizado com base em informações pré-estabelecidas baseadas no perfil do usuário. O que abre a possibilidade de se

agregar aos programas de TV interativa um caráter personalizado, em contraste à idéia original de conteúdo de massa.

1.4 ESTRUTURA DO TRABALHO

No CAPÍTULO 2 é apresentada uma revisão de sobre sistemas de TV Digital, em especial o *International Standard for Digital Television* (ISDTV), o padrão brasileiro de TV Digital.

No CAPÍTULO 3 apresentam-se características e funcionalidades do *middleware* do sistema brasileiro de TV digital. Este capítulo também apresenta considerações sobre os paradigmas de programação procedural e declarativo, suportas pelo *middleware*. As linguagens de programação *Nested Context Language* (NCL), Lua e Java (Java DTV) também são abordadas neste capítulo.

O CAPÍTULO 4 aborda os fundamentos adotados para o tratamento dos diferentes requisitos de aplicações interativas. Neste capítulo, são descritas e analisadas, as abordagens propostas para o tratamento de aplicações interativas dinâmicas, relacionadas semanticamente ao fluxo principal de áudio e vídeo. Ainda nesta parte do trabalho é apresentado um modelo baseado no tratamento de eventos e na estruturação dos dados que compõem a aplicação interativa.

O ambiente de simulação desenvolvido para validação do modelo de aplicações interativas baseadas no tratamento de eventos e na estruturação dos dados da aplicação é descrito no CAPÍTULO 5.

A descrição e análise do estudo de caso da aplicação desenvolvida são realizadas no CAPÍTULO 6.

No CAPÍTULO 7 são apresentadas as conclusões do trabalho.

CAPÍTULO 2

2 TECNOLOGIA DE TV DIGITAL

De acordo com Arnold (2007), o ímpeto inicial para a transição do sistema analógico para o digital foi a possibilidade de conversão entre padrões analógicos. O padrão *National Television Standards Committee* (NTSC) com 525 linhas e 30 imagens/s e o *Phase Alternate Line* (PAL) com 625 linhas e 25 quadros/seg. No domínio analógico esse seria um processo extremamente difícil. Já no domínio digital, embora haja a necessidade de significativo processamento de sinais, seria possível construir componentes de hardware de alto desempenho para execução da tarefa.

No entanto, a digitalização dos sistemas de TV é justificada por uma série de outros benefícios disponíveis a usuários e provedores do serviço, tais como: (i) o melhor aproveitamento da largura de banda disponível, possibilitando a transmissão de múltiplos canais dentro da mesma faixa de frequência ocupada pelo sistema analógico ou a transmissão do conteúdo televisivo em diferentes níveis de resolução; (ii) a possibilidade de transmissão de dados, dando oportunidade para a inclusão de uma série de serviços interativos atrelados ou não aos programas; (iii) imunidade ao ruído, permitindo transmissões de qualidade a longas distâncias através de múltiplos repetidores e; (iv) o custo reduzido e alta confiabilidade dos componentes digitais.

A seguir é apresentada uma breve revisão sobre sistemas de TV Digital.

2.1 PADRÕES INTERNACIONAIS DE TV DIGITAL

Sistemas de TV Digital são concebidos de acordo com condições, peculiaridades e objetivos de seus proponentes. Com o Sistema Brasileiro de TV Digital isto não foi diferente. No entanto, o fato de sua especificação ter ocorrido após a implantação de outros sistemas,

possibilitou a análise de sistemas de TVD maduros implementados em outros países. De acordo com a análise destes outros padrões de TVD, foi possível determinar qual padrão existente se adequava melhor aos requisitos do sistema brasileiro e a partir de então propor as alterações necessárias ao sistema de base, que resultaram na especificação do *International Standard for Digital Television (ISDTV)*, o padrão brasileiro de TVD.

Os padrões internacionais existentes, nos quais o comitê do SBTVD pôde basear-se para análise, e posterior definição do sistema brasileiro foram os seguintes:

- O padrão americano: *Advanced Television Systems Committee (ATSC)*.

Em substituição ao padrão analógico NTSC, foi especificado o padrão americano de TV digital denominado ATSC, adotado nos EUA e Canadá. O padrão engloba uma variedade extensa de subsistemas requeridos para originar, codificar, transportar, transmitir e receber áudio, vídeo e dados por meio de sistemas de sistemas de radiodifusão e cabos.

Para a compressão de vídeo foi adotado o padrão *Moving Picture Experts Group – 2 (MPEG-2)* e para o áudio digital foi especificado um padrão próprio para compressão utilizando o sistema Dolby AC-3. A multiplexação obedece a especificação MPEG-2 com pacotes de 188 bytes e para modulação do sinal foi utilizado o 8- *Vestigial Sideband (8-VSB)*. Vídeo, áudio e dados são transmitidos através dos mesmos canais de 6 MHz disponíveis para a TV analógica a uma taxa útil de 19.28 Mb/s (ATSC, 2010).

Em suma, o padrão americano é caracterizado pela alta definição, porém não apresenta preocupação com a mobilidade e a interatividade (MONTEZ, 2005).

- O padrão europeu: *Digital Video Broadcasting (DVB)*.

O consórcio europeu para especificação do padrão de TV digital com transmissão via radiodifusão, *European Telecommunications Standards Institute*, definiu o DVB com base em um sistema de transmissão de dados, vídeo e áudio digital comprimido, em um fluxo de transporte MPEG-2, utilizando multiplexação *Coded Orthogonal Frequency Division Multiplex (COFDM)* com modulação *Quadrature Phase-shift Keying (QPSK)*, *Quadrature Amplitude Modulation (16QAM)* ou 64QAM.

O padrão *Digital Video Broadcasting – Terrestria* (DVB-T) especifica um sistema projetado para transmitir vídeo digital de alta qualidade, áudio digital e dados através dos canais de 7 e 8 MHz existentes com taxas de dados que vão de 4.98 a 31.67 Mb/s.

Requisitos únicos das redes e estações de transmissão européias foram, em parte, responsáveis pela escolha do COFDM como sistema de modulação. Na Europa redes de frequência única (*Single-Frequency Networks - SFN*) são utilizadas extensivamente para possibilitar a utilização mais eficiente dos canais disponíveis. Elas são sincronizadas para terem o mesmo sinal de referência e temporização de banda base. Um receptor sintonizado nesse canal pode receber sinais de uma ou mais estações simultaneamente, cada um com um atraso diferente. Nessas situações os intervalos de guarda e a equalização embutidos no sistema COFDM facilitam a recepção efetiva (DVB, 2010).

O padrão europeu especifica uma transmissão hierárquica dividindo o fluxo de dados de entrada em dois fluxos separados, de baixa e alta prioridade, cada qual podendo ser transmitido com diferentes formas de codificação no canal e diferentes técnicas de modulação nas sub-portadoras. Isso permite ao difusor definir relações de taxa de bit e robustez para os dois fluxos.

Entre os padrões de TV digital, o DVB é o mais difundido sendo adotado em toda Europa, Oceania, parte da Ásia, África e do continente Sul-Americano, conforme visto na Figura 2.

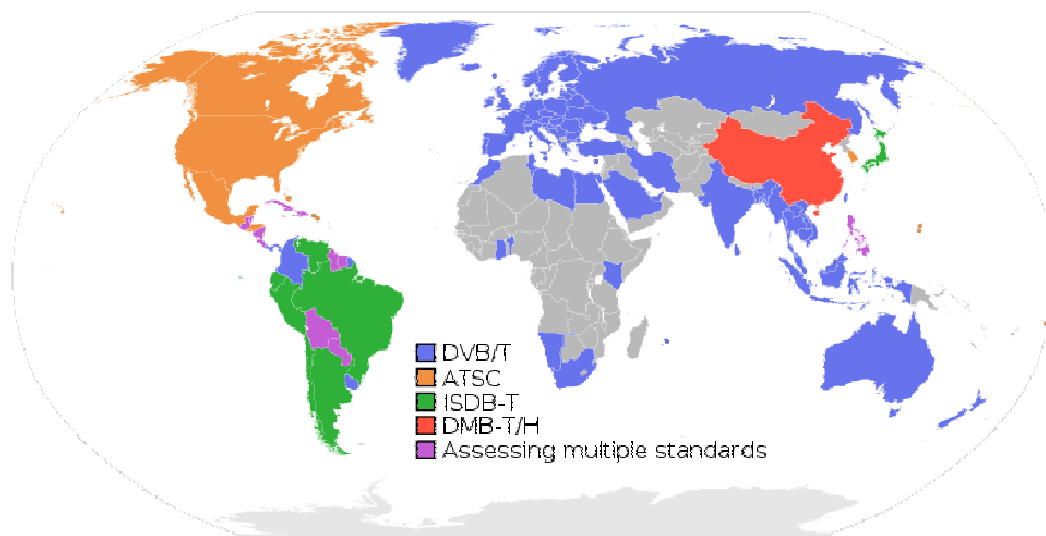


Figura 2 - Padrões Internacionais de TVD

Em suma, o padrão europeu define um sistema com suporte a múltipla programação, interatividade e novos serviços (MONTEZ, 2005).

- O padrão japonês: *Integrated Services Digital Broadcasting (ISDB)*.

O padrão japonês de TV digital, desenvolvido pelo *Digital Broadcasting Experts Group (DiBEG)*, é em muitos aspectos semelhante ao DVB com uma diferença chave no que diz respeito a transmissão por banda segmentada OFDM. Essa abordagem de segmentação de dados permite flexibilidade na alocação de largura de banda a vários serviços, incluindo dados, rádio, TV em definição padrão e em alta definição.

O padrão para radiodifusão terrestre, o *Integrated Services Digital Broadcasting – Terrestrial (ISDB-T)*, especifica um sistema projetado para transmitir através dos canais existentes de 6, 7 e 8 MHz, com taxas de dados de 3.56 a 30.98 Mb/s.

Assim como nos outros padrões mundiais de TV digital o sinal digital utiliza o sistema de transporte MPEG-2 com pacotes de dados de 188 bytes. A carga útil do pacote pode incluir vídeo digital, áudio digital, texto, gráficos e dados. Para possibilitar a transmissão hierárquica foi definido um sistema de controle de transmissão e multiplexação. A segmentação de banda permite que o fluxo de dados seja remultiplexado e organizado em grupos de dados, cada um representando parte, ou o todo, de programa ou serviço. Após a codificação do canal, esses

grupos de dados tornam-se segmentos OFDM, cada um deles ocupando 1/14 da largura de banda do canal.

Os segmentos OFDM dão possibilidade de transmissão simultânea de sinais de alta e baixa definição, múltiplos sinais de definição padrão, áudio e dados. Como é possível definir no receptor que, ou quais, segmentos OFDM serão decodificados, o sistema torna-se então bastante flexível no atendimento do serviço em qualidade e condições diversas (Figura 3) (ISDB, 2010). Desta forma o sistema torna-se compatível a dispositivos móveis, dispositivos com baixa resolução ou capacidade de processamento, e dispositivos de alta definição.

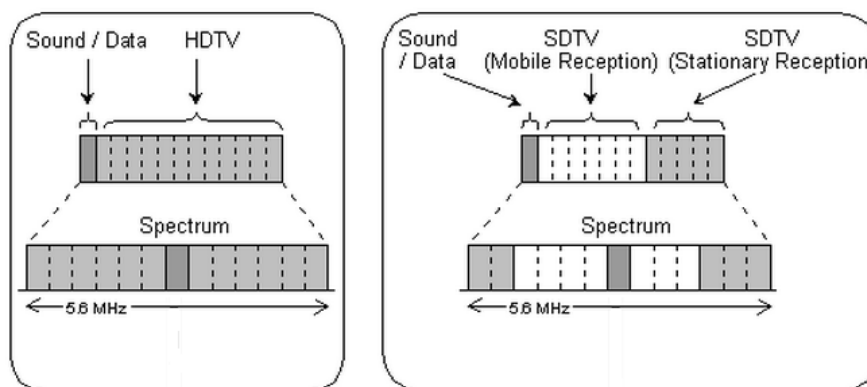


Figura 3 - Transmissão Hierárquica e Recepção Parcial ISDB

O padrão japonês é caracterizado pela alta definição e pelo suporte a mobilidade (MONTEZ, 2005).

2.2 O PADRÃO BRASILEIRO

O padrão brasileiro de TV digital, o *International Standard for Digital Television Terrestria* (ISDTV), foi baseado no modelo japonês, com modificações no método de compressão de vídeo, e no *middleware* (MENDES, 2007). Para compressão de vídeo foi adotado o padrão H.264, uma técnica mais recente e eficiente que o MPEG-2. O H.264 possibilita uma sensível redução da taxa de bits mantendo a qualidade da imagem e permitindo o uso mais eficiente do espectro de transmissão. Para o *middleware* foi definida a especificação e implementação de uma solução nacional capaz de atender aos requisitos únicos do sistema como a questão de inserção social e digital da população, além de uma

série de outros requisitos, abordados na seção 2.2.1. A especificação deste *middleware* foi denominada Ginga.

O comitê do SBTVD, formado por especialistas representantes da indústria, redes de TVs, cientistas, representantes da população e do governo, foi incumbido da definição do modelo internacional que serve de base para a especificação do modelo brasileiro, assim como a especificação do sistema de transporte, recepção, apresentação e interatividade do modelo nacional de TV digital.

2.2.1 As Escolhas do SBTVD

Alguns dos fatores determinantes na escolha dos elementos componentes do sistema brasileiro, conforme norma da Associação Brasileira de Normas Técnicas (ABNT), foram os seguintes (ABNT, 2008):

- a) Multiplexação/Modulação: Dentre as tecnologias adotadas nos padrões de TV digital já estabelecidos, a especificação de um sistema baseado no sistema japonês, o ISDB-T, foi pautada pelo diferencial, de entre os sistemas disponíveis, ser o de melhor suporte a mobilidade. O ISDB-T é uma evolução do sistema europeu, o DVB-T, melhorado em aspectos como melhor desempenho na presença de interferências concentradas, utilização de 13 segmentos independentes dentro da banda de 6MHz com possibilidade de utilização de modulação PSK, QAM e suas variantes (QPSK, DQPSK, 16QAM e 64QAM);
- b) Compressão de Vídeo: Para o esquema de compressão de vídeo a ser utilizado no SBTVD foi adotado o MPEG-4 com codificação H.264, uma solução mais eficiente que o seu predecessor, o MPEG-2. Todos os outros padrões de TV digital adotam o padrão MPEG-2, pois quando da definição dos respectivos padrões de compressão de vídeo digital este era a melhor solução disponível;
- c) Compressão de Áudio: Para compressão e codificação do áudio digital adotou-se o MPEG-4 - AAC (Advanced Áudio Coding), um padrão de compressão com perdas, projetado para ser o sucessor do padrão MP3, atingindo melhor qualidade de áudio que

o seu predecessor em diversas taxas de bit. O padrão brasileiro de TV Digital é compatível com o áudio 2 canais (Stereo) e 5.1 canais (Dolby Digital);

- d) Transporte: Para possibilitar a multiplexação de vídeo, áudio e dados digitais com sincronismo para reprodução foi adotado o protocolo de comunicação especificado pela recomendação H.222 do *International Telecommunication Union - Telecommunication Standardization Sector* (ITU-T), também conhecida como Parte 1 da especificação MPEG-2. O fluxo de transporte MPEG2 oferece funcionalidades de correção de erros para transporte sobre mídias não confiáveis encapsulando fluxos elementares de informação contidos em pacotes em um container de transporte;
- e) Middleware: Devido aos requisitos de inclusão social e fortalecimento da industrial e academia local, definidos no decreto presidencial no. 4.901, fez-se necessária a especificação de um sistema intermediário de software (*middleware*) com funcionalidades voltadas para os aspectos únicos da sua aplicação. Tal sistema tem como função básica, permitir a interoperabilidade entre o conteúdo digital fornecido pela emissora de TV (programas interativos e dados) e as diferentes arquiteturas de *hardware* dos dispositivos de decodificação do sinal digital (implementados em set-top-boxes ou nos próprios aparelhos televisores). Para o sistema brasileiro de TV digital foi então especificado o *middleware* denominado GINGA, um software desenvolvido com tecnologia nacional voltado para a solução dos desafios de implementação requisitados no decreto que constituiu o sistema brasileiro de TV digital.

2.3 CONSIDERAÇÕES FINAIS

O padrão brasileiro de TV digital teve a vantagem de ter sido especificado após a implementação dos grandes padrões internacionais. Desta forma a análise desses modelos, já operacionais, possibilitou a especificação de um sistema moderno, construído sobre a experiência dos sistemas existentes. As principais características do ISDTV são o suporte a alta definição, flexibilidade, mobilidade e interatividade. Os atributos do sistema brasileiro são hoje, referencia para adoção e especificação de sistemas de TVD (FORUM, 2010).

CAPÍTULO 3

3 APLICAÇÕES INTERATIVAS PARA TV DIGITAL

A capacidade de transmitir dados de programas junto ao conteúdo televisivo (Figura 4) tradicional tornou possível a convergência de uma série de funcionalidades para o aparelho de televisão (ou decodificador), criando oportunidades para uma série de novos serviços e agregando conteúdo e valor a cadeia de produção de TV.

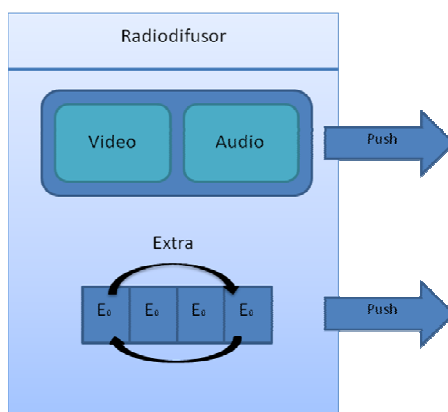


Figura 4 - Transmissão de vídeo, áudio e dados extras

Os dados transmitidos junto ao fluxo de áudio/vídeo podem resultar em funcionalidades de baixa interação, com interação local (i.e., interagir com outros dados transmitidos junto com o fluxo principal), ou interação global, o que só é possível através da disponibilidade de um canal de comunicação, o canal de interatividade ou canal de retorno. Os novos serviços agregados vão desde informações sobre a grade de programação da emissora a sinopse de programas, jogos, previsão do tempo, e-mail, compras on-line, programas educativos interativos, navegação na Web, possibilidade de compor diferentes visões de um mesmo programa baseando-se no perfil do usuário, etc.

O suporte a essas funcionalidades se dá por meio da definição de um *middleware* responsável por compatibilizar aplicações e dispositivos, oferecendo recursos e serviços às aplicações de maneira uniforme.

3.1 MIDDLEWARES

Middleware é um neologismo para a camada de software localizada entre as aplicações e o sistema operacional (Figura 5). Suas finalidades são esconder das aplicações detalhes de implementação das camadas inferiores, possibilitar maior interoperabilidade entre aplicações e sistemas heterogêneos (sistemas operacionais e hardware), oferecer serviços para aplicações e suporte para o desenvolvimento de programas.



Figura 5 - Arquitetura de camadas de um sistema computacional de TVD

No contexto de TV Digital, o *middleware* vem a ser o *software* que controla suas principais facilidades (grade de programação, menus de opção, configuração de perfis, etc.), inclusive a possibilidade de execução de aplicações, dando suporte à interatividade (RNP, 2004).

Visando facilitar o desenvolvimento das aplicações, garantir a compatibilidade das mesmas com as diversas arquiteturas existentes e ocultar particularidades e diferenças do hardware e sistema operacional, as principais propostas de sistemas de TV digital trazem, além da especificação das interfaces de meio físico e da codificação dos sinais de transmissão, propostas de *middlewares* para Televisão Digital. Dentre as principais propostas de

middleware para TVDI podem ser citados: o *Multimedia Home Platform* (MHP) (MHP, 2010) para o sistema DVB, o *Advanced Common Application Platform* (ACAP) (ACAP, 2005) para o sistema ATSC e o *Association of Radio Industries and Business* (ARIB-STD 24) (ARIB, 2004) para o sistema ISDB. Esses *middlewares* oferecem mecanismos para carregar o código dos aplicativos e executá-los, além de serviços que permitem:

- a) gerência de aplicações e recursos do *middleware*;
- b) comunicação entre aplicações e processos;
- c) processamento (decodificação e exibição) de mídias;
- d) sincronismo;
- e) seleção, acesso e interpretação de fluxos elementares e informações de serviço;
- f) tratamento de eventos e elementos áudio-visuais de interface com o usuário;
- g) armazenamento, localização e recuperação de dados locais; e
- h) controle do ciclo de vida das aplicações.

Outras funcionalidades podem ser oferecidas de acordo com o *middleware* e com os recursos presentes no terminal de acesso, ie., Set-Top-Box (STB), tais como sistemas de acesso condicional, acesso a dados remotos, perfis de usuários, acesso ao fluxo de áudio e vídeo principal sendo exibido (LEITE, 2005).

3.2 O GINGA

Ginga é o *middleware* do padrão brasileiro de TV digital. Comparado com outros sistemas *middleware* concebidos para outros padrões de TV digital possui algumas funcionalidades inovadoras. Essas funcionalidades são oriundas da sinergia resultante da conjunção de dois projetos: (i) FlexTV, o *middleware* procedural de referência do SBTVD e (ii) MAESTRO, o *middleware* declarativo de referência do SBTVD (SOUZA, 2007).

3.2.1 Arquitetura

O Ginga possui suporte a aplicações procedurais e declarativas e em sua especificação dois subsistemas lógicos para execução de aplicações são implementados. O subsistema lógico Ginga-NCL implementa o ambiente de execução e apresentação de documentos

declarativos escritos em linguagem NCL (Nested Context Language) (NCL, 2010). O subsistema lógico procedural, denominado Ginga-J, implementa um ambiente de execução e apresentação de programas baseados na tecnologia Java. Embora o suporte a aplicações declarativas e procedurais esteja implementado em módulos separados do sistema, aplicações Ginga não são necessariamente puramente declarativas ou procedurais. Funcionalidades comuns aos dois subsistemas são implementadas por um componente denominado Ginga-Core (SOUZA, 2007).

O Ginga-Core (Figura 6) é o responsável por prover necessidade comuns aos dois subsistemas como decodificação e apresentação de conteúdos PNG, JPEG, MPEG e outros formatos, transporte de conteúdo via fluxo de transporte MPEG-2, canal de interatividade e suporte ao modelo de exibição conceitual do ISDTV.

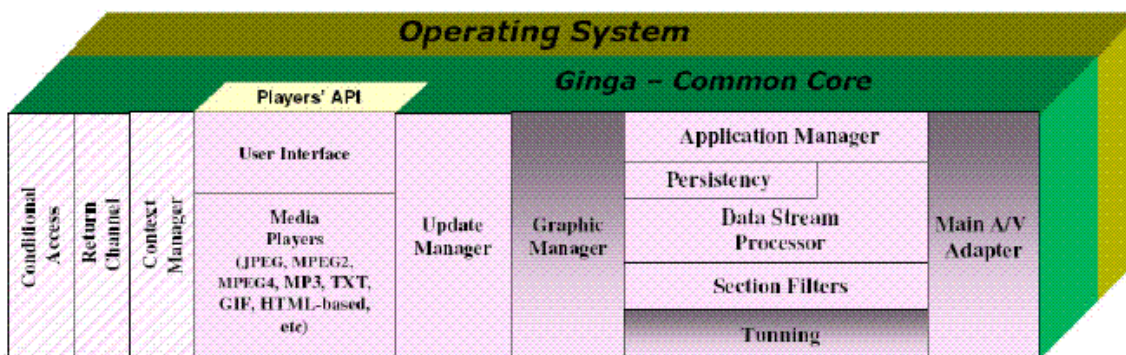


Figura 6 - Ginga-Core

O subsistema Ginga-NCL (Figura 7) é o responsável pela execução e apresentação de documentos escritos em NCL, uma linguagem de aplicação XML, baseada no modelo conceitual *Nested Context Model* (NCM) (SOARES, 2005). O modelo NCM especifica documentos hipermídia com relações de sincronismo espacial e temporal entre os objetos de mídia (imagens, texto, áudio, vídeo, etc.).

Neste mesmo subsistema há ainda um módulo Extended Hiper Text Markup Language (XHTML), que inclui um interpretador de folha de estilo (CSS) e ECMAScript (ECMA, 1999), e uma *engine* Lua para suporte a aplicações desenvolvidas na linguagem de script Lua (LUA, 2010).

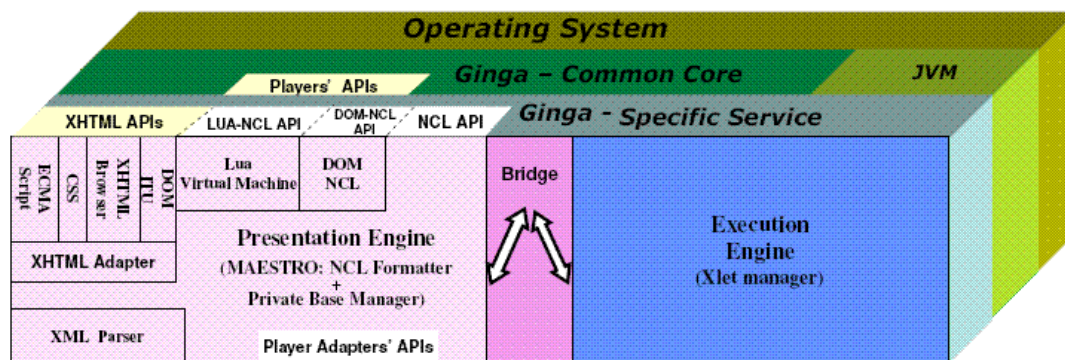


Figura 7 - Ginga-NCL

O subsistema Ginga-J (Figura 8) é a parte procedural do *middleware* Ginga. Este subsistema baseada em tecnologia Java, foi o último componente do *middleware* a ser especificado, devido a questões referentes ao pagamento de *royalties* para a utilização de componentes *Globally Executable MHP* (GEM). O GEM é constituído por um conjunto de componentes (e funcionalidades) que servem como base para o desenvolvimento de aplicações procedurais para os *middlewares* europeu (MHP), americano (ACAP) e japonês (ARIB).

Para substituir os componentes GEM, foi então especificado um conjunto de API's denominado Java DTV. O Java DTV é um pacote de componentes Java, para a confecção de aplicações iTV livre de *royalties*, e foi especificado por meio de uma parceria entre a *Sun Microsystems* (SUN, 2010_) e o SBTVD. Conforme pode ser observado na representação da arquitetura Java DTV (Figura 8), além do suporte às API's Java DTV é previsto o suporte a API's próprias do SBTVD e parte das API's nativas do ARIB, *middleware* do sistema japonês no qual o sistema brasileiro foi baseado.

Recentemente¹, a especificação Ginga-J: *Ambiente de execução de aplicações procedurais*, foi aprovada como norma ABNT registrada sob o identificador 15606-4:2010 (ABNT, 2010).

¹ A Norma ABNT NBR 15606-4:2010 foi publicada em 13/04/2010 e passa a ser válida a partir de 13/05/2010.

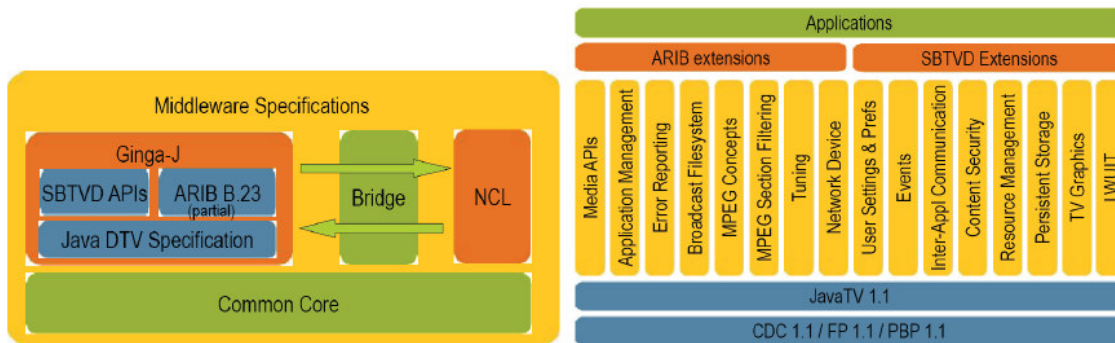


Figura 8 - Arquitetura Ginga-J

3.3 INTERATIVIDADE

Trazer a interatividade para a TV implica na troca do paradigma tradicional, no qual o usuário assiste passivamente o que as emissoras de TV definem em sua programação, para um paradigma no qual o usuário interage com o conteúdo televisivo.

O conceito de TV interativa estende as funcionalidades disponíveis para o usuário (ligar e desligar a TV, mudar de canal, etc.). Dessa forma, a TV passa a ter um funcionamento semelhante a um computador, possibilitando ao telespectador (ou usuário) realizar compras, ler e-mails, personalizar a apresentação do conteúdo televisivo baseado em seu perfil e mais uma infinidade de funcionalidades ainda não exploradas dentro do modelo de negócios tradicional da TV (Figuras 9 e 10).



Figura 9 - Serviço público jurídico, Reino Unido



Figura 10 - Seleção interativa de áudio

Contudo, a oferta de serviços interativos é dependente da infraestrutura computacional disponível ao usuário. Esta estrutura é formada pelo decodificador (presente na forma de um *set-top-box*, ou interno ao próprio televisor), pela emissora de conteúdo e pelo provedor de serviços (no caso de dependência de um meio de comunicação, ou canal de interatividade, para que os usuários acessem os serviços interativos).

Embora aplicações interativas de TV digital já estejam em funcionamento em diversos países, questões relativas à interação usuário-computador, como o melhor aproveitamento do tamanho da tela, o uso da interatividade via controle remoto, o fato de a TV proporcionar uma experiência coletiva ou, ainda, de possuir um único “controlador” de interatividade (o usuário que detém o controle remoto) são ainda objeto de diversos estudos (PICCOLO, 2006) (CESAR, 2009).

3.4 APLICAÇÕES INTERATIVAS

De acordo com (MARQUES, 2008), aplicações para TV Digital podem ser classificadas em três níveis de interatividade: baixa, média e alta. As aplicações de baixa interatividade são aquelas que não dependem de um canal de comunicação, como por exemplo, a exibição da grade de programação da emissora, ou a previsão do tempo para a região dentro da faixa de cobertura de transmissão. As aplicações de média interatividade são aquelas que fazem uso do canal de comunicação, mas, no entanto, não possuem nenhuma espécie de sincronismo temporal ou espacial com o conteúdo apresentado, como por exemplo, uma cliente de e-mail que pode ser acessado a qualquer momento, independente do programa que esteja sendo assistido. O terceiro nível de interatividade trata das aplicações que, além de requisitarem o uso do canal de comunicação, ainda possuem dependência espaço-temporal com o conteúdo televisivo (i.e., relação com o conteúdo semântico de áudio e vídeo apresentados). Aplicações de alta interatividade podem ser ilustradas pela possibilidade de, durante um jogo de futebol ao vivo, o usuário solicitar um *replay* de um lance polêmico, poucos segundos após o ocorrido. Outro exemplo aparece na solicitação da exibição de um mapa do circuito de corrida com a posição dos primeiros colocados naquele exato instante.

Para dar suporte ao funcionamento de aplicações interativas, dados e objetos de mídia (imagens, áudios, vídeos, textos, etc.), precisam estar disponíveis no dispositivo decodificador junto com o fluxo principal (áudio e vídeo sincronizados) do programa. Esses dados, a depender da sua natureza, podem ser enviados pela própria emissora de TV, junto ao conteúdo televisivo tradicional, via canal de interatividade ou acessados, de forma alternativa, como por exemplo, através de uma porta USB (Figuras 3.3A e 3.3B). A natureza dos arquivos de dados solicitados pela aplicação também pode ter impacto na forma como eles serão transmitidos ao usuário. A grade de programação de uma emissora (EPG – *Electronic Program Guide*), por não conter dados pessoais do usuário e não ser composta por arquivos extensos, é um exemplo de aplicação tipicamente enviada pelos radiodifusores junto com a programação usual de uma emissora. A visualização de mensagens de e-mail, devido ao seu caráter pessoal, seria um exemplo de aplicação com dados enviados por intermédio do canal de interatividade. Outra diferença em relação às duas formas de envio de dados ao usuário é o fato do canal de interatividade possibilitar que o usuário receba apenas as informações solicitadas, enquanto que os dados recebidos via emissora são enviados mesmo sem solicitação.



Figura 11 - (A) Visão posterior do SBT Zinwell ZBT-620A. No detalhe, conector USB. (B) Seleção do dispositivo USB para acesso a dados no armazenamento externo

3.5 O CARROSSEL DE DADOS

O carrossel de dados, ou carrossel de objetos, é o mecanismo responsável por enviar de forma cíclica programas e dados multiplexados ao conteúdo televisivo padrão no fluxo de transporte MPEG-2 (Figura 12). O carrossel funciona como um disco virtual que armazena dados e aplicativos a serem disponibilizados aos usuários. Através do carrossel de objetos torna-se possível remontar no decodificador a estrutura de diretório da aplicação e seus dados da maneira em que se encontravam na fonte da transmissão.

O envio cíclico de conteúdo permite ao telespectador o recebimento de programas e dados mesmo que o canal seja sintonizado após o início da transmissão do programa interativo. Para aplicações disponíveis localmente, por exemplo, a apresentação de fotos armazenadas em um pendrive conectado à porta USB do decodificador, não é necessário o emprego do carrossel. Assim como em outras aplicações que utilizam dados provenientes do canal de interatividade (MARQUES, 2008).

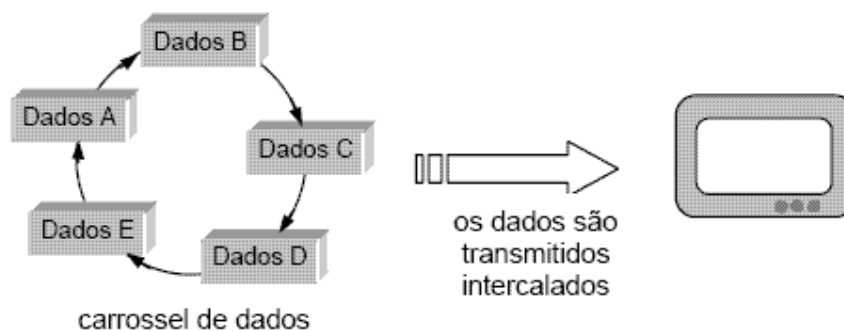


Figura 12 - Carrossel de Dados

3.6 DESENVOLVIMENTO DE APLICAÇÕES INTERATIVAS: CONCEITOS

Segundo Marques (2009a), a construção de *softwares* para TVDi deve ser realizada respeitando as particularidades do ambiente de TV. Entre as principais particularidades pode-se citar:

- a) Os *softwares* são parte de um programa de TV tradicional e esses, por sua vez, tem formato e contexto próprios.
- b) Utilizam como interface o receptor de TV que é de uso tradicionalmente coletivo.
- c) Exigem infra-estrutura de transmissão e componentes de *software/hardware* adequados ao seu funcionamento.
- d) Modificam os programas de TV tradicionais de forma a capacitá-los a lidar com diferentes níveis de interatividade e com uma organização do conteúdo não linear.

Dentro do contexto de desenvolvimento de aplicações interativas para TV, estas particularidades influem na definição da linguagem utilizada na implementação da solução. Esta definição está fortemente ligada à natureza da aplicação e envolve questões que incluem o prazo de entrega, o custo, o tempo de vida da aplicação, etc.

Linguagens declarativas são mais intuitivas e, portanto, mais fáceis de utilizar, o que dispensa o emprego de um especialista em programação para a confecção de aplicações. Elas são adequadas a aplicações com foco no objetivo específico para o qual a linguagem foi desenvolvida. Neste caso, a definição de restrições espaciais e temporais (sincronismo), entre

as mídias que compõem a aplicação (mais detalhes sobre o paradigma declarativo encontram-se na seção 3.6.1.1.).

O uso de linguagens procedurais é mais apropriado em aplicações que não são orientadas ao foco das linguagens declarativas (seção 3.6.2.1.). Linguagens procedurais normalmente requerem o emprego de um especialista em programação. Como na TV digital os dois tipos de aplicações devem coexistir, é conveniente que o receptor integre o suporte tanto ao paradigma de programação declarativa quanto ao procedural. A seguir, serão analisados os diferentes paradigmas de programação (declarativo e procedural) disponibilizados no Ginga.

3.6.1 Paradigma Declarativo

De acordo com Soares (2006):

[...] nas linguagens declarativas, o programador fornece apenas o conjunto das tarefas a serem realizadas, não estando preocupado com os detalhes de como o executor da linguagem (interpretador, compilador ou a própria máquina real ou virtual de execução) realmente implementará essas tarefas. Em outras palavras, a linguagem enfatiza a declaração descritiva de um problema ao invés de sua decomposição em implementações algorítmicas, não necessitando, em geral, de tantas linhas de código para definir uma certa tarefa.

A necessidade de sincronização em um sistema de TV digital está presente mesmo em sua aplicação mais primária: a exibição sincronizada do fluxo de vídeo e áudio que compõem o fluxo principal de um programa de TV. As aplicações para TV digital, muitas vezes, devem lidar com a sincronização espacial e temporal de diferentes tipos de mídia, além dos objetos de vídeo e áudio que compõem o fluxo principal. Considere, como exemplo, uma aplicação interativa disponibilizada ao usuário por um determinado período (por exemplo, 5 segundos), no momento em que um ator entra em um carro. Ao se iniciar o período onde a interatividade é oferecida ao usuário, um vídeo-propaganda de uma montadora de automóveis deve ser exibido. Note que, nesse caso, o sincronismo foi baseado em um evento previsível do tempo (o momento de entrada do ator no carro). Entretanto, as aplicações multimídia devem prever também o tratamento de eventos imprevisíveis no tempo, como a interação do usuário.

O foco dos subsistemas declarativos associados aos três principais sistemas de TV digital (europeu, americano e japonês) é a interatividade. Isto é, eles têm como objetivo facilitar o desenvolvimento das aplicações interativas na TV. Uma característica comum a estes *middlewares* é que todos adotaram derivações da linguagem HTML para dar suporte a aplicações declarativas. Esse foco é, no entanto, bastante restritivo. Para contornar suas restrições, as derivações de HTML, nos sistemas de TV, permitem a inclusão de objetos especificados na linguagem ECMAScript (ECMA, 1999). Qualquer outro tipo de sincronismo mais complexo de mídias, ou procedimentos algorítmicos, terá de ser descrito através de scripts, ou seja, de forma mais complexa e mais longe dos objetivos do projetista da aplicação (SOARES, 2006).

3.6.1.1 A Parte Declarativa do Ginga: Ginga-NCL

Para o padrão brasileiro de TV digital foi especificada uma linguagem declarativa baseada no modelo *Nested Context Model* (NCM), denominada *Nested Context Language* (NCL). A linguagem NCL visa facilitar o processo de desenvolvimento de aplicações interativas não-lineares, que constituem a maioria dos programas interativos (SOARES, 2005). O termo programa não-linear vem em contraposição à forma seqüencial – portanto, linear – que caracteriza os programas para a TV analógica. Nesses últimos, existe um e apenas um caminho seqüencial de exibição. Ao contrário, os programas não-lineares são compostos de múltiplas cadeias de exibição, algumas exibidas em paralelo e outras como alternativas (ou seja, ou uma cadeia ou a outra) que dependem da escolha do usuário, do terminal onde o programa será exibido, da região onde o telespectador está inserido, etc. O sincronismo de mídias em sua forma mais ampla é o foco da linguagem NCL, que não define nenhum objeto de mídia, mas apenas o vínculo que mantêm esses objetos semanticamente juntos em uma apresentação multimídia, esta linguagem torna-se então opção preferencial no desenvolvimento da maioria das aplicações de TV digital, pois seu foco está no relacionado com o foco de grande parte das aplicações para TVD.

```

<head>
  <connectorBase>
    <causalConnector id="onBeginDelayStart">
      <simpleCondition role="onBegin"/>
      <simpleAction role="start" delay="5s"/>
    </causalConnector>
  </connectorBase>
  <regionBase>
    <region id="regiaoTV" width="100%" height="100%">
      <region id="regiaoVideo" left="10%" top="10%" width="80%" height="80%">
      <region id="regiaoImagem1" left="50" top="200" width="400" height="181"/>
    </region>
  </regionBase>
  <descriptorBase>
    <descriptor id="descriptorVideo" region="regiaoVideo"/>
    <descriptor id="descriptorImagem1" region="regiaoImagem1"/>
  </descriptorBase>
</head>
<body>
  <port id="inicio" component="video"/>
  <media id="video" src="video1.mpg" descriptor="descriptorVideo"/>
  <media id="imagem1" src="imagem1.jpg" descriptor="descriptorImagem1"/>
  <link id="iniciaImagem" xconnector="onBeginDelayStart">
    <bind component="video" role="onBegin"/>
    <bind component="imagem1" role="start"/>
  </link>
</body>

```

Figura 13 - Aplicação interativa escrita em NCL

A Figura 13 mostra um exemplo de um simples programa interativo em NCL. O código NCL é dividido em cabeçalho (*head*) e corpo (*body*). No cabeçalho, são definidos as regiões, os descritores e as regras utilizadas pelo programa. No corpo, são definidos o contexto, os nós de mídia, elo e outros elementos que definem o conteúdo e a estrutura do programa.

3.6.1.2 A Linguagem de *Script* Lua

De acordo com Soares (2008), aplicações declarativas necessitam de um engenho de apresentação (i.e., navegador) e são apresentadas e implementadas de forma similar a uma página HTML, podendo conter scripts e/ou folhas de estilo. Para prover suporte a geração dinâmica de conteúdo (ou seja, em tempo de execução) o módulo Ginga-NCL agrega o suporte a execução de scripts em linguagem Lua. Lua é uma linguagem de programação poderosa e leve, projetada para estender aplicações. Essa linguagem combina sintaxe simples para programação procedural com poderosas construções para descrição de dados, baseadas em tabelas associativas e semântica extensível. Lua é tipada dinamicamente e interpretada a partir de *bytecodes* para uma máquina virtual, com gerenciamento automático de memória e coleta de lixo incremental. Essas características fazem a linguagem Lua adequada para

configuração, automação (*scripting*) e prototipagem rápida. A execução de *scripts* lua é realizada através de uma máquina virtual (*engine*), acoplada ao Formatador NCL (SANT'ANNA, 2008). Isso significa que, além de sintaxe e semântica, ela fornece uma API que permite a aplicações NCL trocar dados com *scripts* Lua. É importante também destacar a integração entre Lua e Java, através da biblioteca LuaJava (LUA, 2010), que permite o acesso a qualquer classe Java a partir do código Lua, de forma similar ao que acontece com ECMAScript. Além disso, o LuaJava permite que a manipulação do ambiente de Lua a partir do código Java, tornando-se, assim, parte da “ponte” entre os ambientes declarativo e procedural do *middleware* Ginga (SOARES, 2008).

A máquina Lua está implementada como uma pequena biblioteca de funções C, escritas em ANSI C, que pode ser compilada, sem modificações, sobre todas as plataformas conhecidas. Os objetivos da implementação são simplicidade, eficiência, portabilidade e baixo impacto de inclusão em aplicações (IERUSALIMSKY, 2003).

3.6.2 Paradigma Procedural

Na programação procedural (também chamada de imperativa) deve-se informar ao computador cada passo a ser executado. Pode-se afirmar que, em linguagens procedurais, o programador possui um maior controle do código, sendo capaz de estabelecer todo o fluxo de controle e execução de seu programa. Entretanto, para isso, ele deve ser bem qualificado e conhecer bem os recursos de implementação disponíveis (SOARES, 2006). As linguagens procedurais são bem expressivas e de propósito gerais, porém possuem um custo alto de implementação e seu emprego no desenvolvimento de programas não-lineares eleva consideravelmente a complexidade de implementação da solução.

De acordo com Marques (2009a), o modelo de concepção de produções de TV é fortemente centralizado no gerador de conteúdo. Este modelo dificulta a participação de “agentes”, tais como engenheiros de software e programadores, na cadeia produtiva de conteúdo de TV. Outras dificuldades na aplicação deste modelo seriam a baixa flexibilidade e extensibilidade das aplicações desenvolvidas. Como exemplo, pode ser citada, uma aplicação interativa concebida inicialmente para ser apresentada em um televisor e posteriormente

extendida para um celular (requisito não funcional). Tal extensão pode significar em refazer boa parte do sistema.

Uma saída para o problema ilustrado anteriormente, seria a utilização de um modelo para especificação de programas de TVDi voltados à utilização de componentes de *software*. Neste modelo a “visão” do produtor de conteúdo (*storyboard*) é transformada em um conjunto de componentes e posteriormente em código objeto. O objetivo deste modelo é diminuir a responsabilidade do gerador de conteúdo por meio da descentralização das etapas de produção que estão fora do seu universo de trabalho original.

3.6.2.1 A Parte Procedural do Ginga: Ginga-J

O subsistema Ginga-J é a parte do *middleware* brasileiro responsável pelo processamento de aplicações procedurais escritas na linguagem Java. Ele é composto pela Máquina Virtual Java, o conjunto de API's Java DTV e API's do SBTVD e ARIB.

A grande vantagem de se adotar o Java como tecnologia de base para o subsistema procedural está em seu elevado grau de interoperabilidade, requisito chave para aplicações interativas de TV. Somado a este fato, a popularidade da linguagem Java, contribui para a formação e recrutamento de mão de obra especializada para o desenvolvimento de aplicativos.

3.7 CONSIDERAÇÕES FINAIS

Para oferecer suporte aos requisitos do decreto presidencial que institui o SBTVD, foram especificados uma série de componentes de *software* voltados a provisão de recursos e facilidades para o desenvolvimento, transmissão e apresentação de aplicações interativas. No entanto, o desenvolvimento (e a posterior aceitação) de aplicações interativas envolve uma série de novos desafios aos seus interessados. Do lado produtor de TV convencional, são inseridas questões que envolvem decisões de autoria relacionadas à dinâmica da aplicação interativa e a programação de TV (sincronismo e não-linearidade), a questões de comunicação com os novos integrantes da cadeia de produção (engenheiros de *software*, programadores, etc.). Do lado do responsável pela confecção e manutenção da aplicação interativa, questões

como usabilidade e flexibilidade apresentam-se diante de um novo cenário para apresentação e consumo de aplicações interativas, o televisor. No entanto, a abordagem completa das questões relativas ao desenvolvimento de aplicações interativas foge do escopo deste trabalho. O próximo Capítulo trata de algumas questões relativas aos desafios encontrados no desenvolvimento de aplicações de TVDi, para o caso específico de programas interativos com transmissão ao vivo.

CAPÍTULO 4

4 PROGRAMAS INTERATIVOS DE TV

Aplicações multimídia e hipermídia são usualmente compostas de relacionamentos entre objetos de mídia estática (textos, fotografias) e dinâmica (vídeo, áudio). Esses relacionamentos são definidos por meio de procedimentos que especificam como estes objetos devem ser apresentados (MARQUES, 2009b). Tal apresentação ocorre dentro de requisitos de espaço e tempo pré-definidos. Estes requisitos de definição da estrutura de sincronização (coordenação no espaço e tempo), entre os diferentes tipos de mídia que compõe a aplicação, são de suma importância na modelagem da aplicação (COURTIART, 1996). Logo, o processo de autoria de aplicações multimídia é uma questão que envolve sincronismo em vários níveis (MARQUES, 2009b).

De acordo com Marques (2008), aplicações interativas para TVD podem ser definidas em três grupos: (i) Aplicações sem nenhuma relação com o conteúdo semântico de áudio e vídeo apresentado; (ii) Aplicações relacionadas ao conteúdo semântico do fluxo principal apresentado, porém sem fortes restrições de sincronização e (iii) aplicações com relacionamentos de sincronismo com o fluxo principal, esta última englobando aplicações com conteúdo de mídia previamente conhecido e aplicações com geração dinâmica de conteúdo (em tempo de apresentação).

Para qualquer um dos casos citados, na especificação de sincronismo temporal entre mídias, geralmente são adotados modelos que expressão relacionamentos baseados em entidades como nós, âncoras e *links* (HALASZ, 1994). Dentre estes modelos, podem ser ressaltados os baseados em intervalos, hierarquia, linhas de tempo e eventos (BENBERNOU, 2005). Os modelos temporais permitem definir “quando” apresentar as mídias que compõem a aplicação.

A abordagem usual, para definição da estrutura de apresentação em modelos temporais, requer a definição a priori das mídias que compõem a aplicação e seus relacionamentos.

Logo, naturalmente, autoria e apresentação ocorrem em fases distintas. A descrição dos requisitos de sincronismo que envolvem eventos, restrições e expressões lógicas e temporais, constitui a fase de autoria. Durante a fase de apresentação os controles de sincronismos consistem na aplicação, em tempo de execução, do grupo de regras extraídas dos requisitos de sincronização (MARQUES, 2008).

Essa abordagem, no entanto, não contempla a modelagem de aplicações em que não é possível definir que mídias irão compor a apresentação nem quando os eventos responsáveis pela apresentação das mesmas virão a ocorrer. Um cenário que exemplifica essa problemática inclui programas interativos transmitidos ao vivo, como uma partida de futebol. Neste programa, a configuração dos relacionamentos envolveria elementos desconhecidos a priori, como lances perigosos, faltas, substituições e gols. Tais elementos dinâmicos, embora esperados, não ocorrem em momentos pré-definidos nem se encontram representados em mídias conhecidas. Esta prerrogativa impossibilita a definição de seus relacionamentos com o fluxo principal de áudio e vídeo em tempo de autoria.

4.1 MODELOS PARA EDIÇÃO DE APLICAÇÕES AO VIVO

Para melhor ilustrar as alternativas ao modelo de autoria em que os relacionamentos e mídias que compõem a aplicação interativa devem ser definidos a priori, convém ressaltar a diferença entre a estrutura lógica e a estrutura de apresentação de um documento multimídia/hipermídia.

A estrutura lógica do documento é definida por relacionamentos que, quando interpretados, permitem a construção da estrutura de apresentação. Um exemplo disto seria a organização lógica de uma novela. Tal organização se dá por capítulos, compostos de cenas, que por sua vez são formadas por tomadas de câmera, etc. Já a estrutura de apresentação desta mesma novela, pode ser formada por uma seqüência de tomadas de câmera, de cenas mescladas, cuja estrutura lógica é mentalmente recomposta pelo telespectador (MORENO, 2006).

Uma primeira alternativa voltada para a edição dos requisitos de sincronismo durante a apresentação consiste em entregar ao ambiente de exibição somente a estrutura de

apresentação. Nesta abordagem a estrutura de apresentação é definida no ambiente de autoria e disponibilizada ao ambiente de edição passo-a-passo, durante o decorrer da apresentação. O envio granular da estrutura de apresentação permite então, a alteração dinâmica da apresentação. As desvantagens dessa abordagem são listadas a seguir (MORENO, 2006):

- a) A necessidade de envio de objetos dinâmicos juntos a sua correspondente estrutura de apresentação, acarreta em restrição quanto ao tempo hábil para atendimento dos requisitos de apresentação;
- b) Restrições para a representação de eventos temporais de duração variável ou desconhecida antes da exibição;
- c) A impossibilidade de realizar ajustes na apresentação;
- d) A dificuldade do receptor (no caso um retransmissor) em realizar a edição da apresentação, uma vez que a estrutura lógica da mesma não é conhecida.

Uma segunda alternativa, para dar suporte à modificação dinâmica da estrutura de exibição, consiste em enviar a estrutura lógica do documento ao ambiente de exibição. No ambiente de exibição a estrutura de apresentação é então gerada a partir da estrutura lógica do documento. No entanto, este ambiente de exibição suporta que modificações na estrutura lógica do documento, estabelecidas dinamicamente no ambiente de autoria, impliquem na atualização da estrutura de apresentação no ambiente de exibição. É importante salientar que apenas a estrutura de apresentação é alterada no ambiente de exibição. A estrutura lógica, entretanto, não é atualizada no ambiente de execução. Esta característica não preserva a estrutura lógica, especificada em tempo de autoria, deixando incoerentes as especificações de autoria e exibição (MORENO, 2006).

Outra alternativa, apresentada em (MORENO, 2006), propõe um modelo onde comandos de edição (assim como sua dinâmica de transmissão) são definidos, para aplicações de TV interativa. Estes comandos são enviados até o receptor por meio de metadados em eventos de fluxo DSM-CC (CASEY, 1996) e juntos, comandos e eventos, implicam em alterações em tempo de execução no documento NCL que contém a estrutura da aplicação de TV interativa. Estas alterações mantêm no ambiente de exibição a mesma estrutura lógica definida no ambiente de autoria. A premissa de base para o modelo é de que a mídia, cuja criação depende de eventos dinâmicos, deve ser preparada e transmitida para os receptores antes da transmissão dos comandos de edição. No entanto não faz parte do escopo da

proposta a definição de um modelo formal para estruturar os dados que compõem o conteúdo de mídia criado nem o suporte a aplicações desenvolvidas em abordagens procedurais.

De acordo com o trabalho apresentado em (MARQUES, 2009b), outra abordagem seria especificar que eventos podem ocorrer durante uma apresentação ao vivo, e não quando esses eventos podem ocorrer. Esta abordagem permite o tratamento de relações de sincronismo envolvendo objetos de mídia desconhecidos durante a fase de autoria. Isto é possível a partir da estruturação dos dados que compõem o conteúdo de mídia e da definição de ações que devem ser executadas para possibilitar o tratamento de eventos ocorridos (ou gerados) em tempo de execução. A apresentação mais detalhada do modelo proposto encontra-se a seguir.

4.1.1 Modelo baseado em eventos

A representação de objetos de mídia, tais como vídeo e áudio, e objetos do mundo real que aparecem no vídeo e áudio, são requisitos para modelos de conteúdo de TV (KENT, 2001). Estes modelos também devem permitir a representação de conteúdo de vídeo e áudio em termos de interações entre objetos e eventos. Em programas de TV interativa, esse conteúdo agrupa vídeo, áudio e dados extras. Estes dados extras incluem (MARQUES, 2008):

- a) Dados extras, relacionados semanticamente ao programa de TV, porém sem fortes restrições de tempo e com a dinâmica do vídeo. Por exemplo, em um jogo de futebol, dados contendo informações sobre os times (elenco, escalação, esquema tático) não possuem relação à dinâmica do vídeo, podendo ser exibidos a qualquer momento. Neste caso o nó de mídia e seus relacionamentos podem ser definidos em tempo de autoria;
- b) Dados extras com dependência dinâmica e temporal ao programa de TV. Em um jogo de futebol, a disponibilização do replay de um gol, é um exemplo de conteúdo dinâmico que embora previsível, não possui um momento definido para ocorrer. Neste caso o nó de mídia e seus relacionamentos devem ser definidos em tempo de apresentação.

Para prover suporte a dados extras com dependência dinâmica e temporal ao conteúdo principal, a abordagem descrita em (MARQUES, 2009b) é focada na estruturação dos dados extras baseada em objetos, tanto no difusor quanto no receptor do programa interativo. A abordagem propõe associar o conteúdo extra difundido com objetos que o representem e transmiti-los, ou ao menos torná-los acessíveis ao receptor (por exemplo, por meio de uma conexão *Web*). O processamento dos dados contidos nos objetos pode identificar a ocorrência de tipos específicos de eventos, e disparar, caso necessário, uma ação. O tratamento dado aos eventos segue o padrão descrito em (ANICIC, 2009). A abordagem é flexível na maneira de transmitir os objetos, que pode seguir o modelo *pull* (quando solicitado pelo aparelho receptor a um agente externo. i.e., *Web*), ou *push* (quando transmitido pelo difusor junto ao fluxo principal de áudio e vídeo).

O modelo proposto em (MARQUES, 2009b) provê informações para organizar informações oriundas de um programa interativo de TV ao vivo baseado em eventos, incluindo tipos de eventos (faltas, gols, lances perigosos), seus atributos (autor do gol, time, tempo de partida no momento do gol), etc. Do ponto de vista estrutural, no mesmo exemplo, o conteúdo multimídia difundido é modelado em notação orientada a objeto.

4.2 ESTRUTURAÇÃO DE CONTEÚDO INTERATIVO

Nesta abordagem, dados estruturados definem o modelo que abstrai o conteúdo interativo de TV (principal e extra), de maneira orientada a objetos. Para melhor entender a abordagem orientada a objetos, do modelo de aplicações interativas de TV, algumas definições são apresentadas a seguir:

- a) *Main Content* (MC) – Conteúdo principal do programa. O fluxo de áudio e vídeo comum a um programa de TV;
- b) *Broadcast Extra Content* (BEC) – Conteúdo extra difundido para os receptores. Formado pelos objetos que estruturam a aplicação interativa do programa;
- c) *Receiver Extra Content* (REC) – Formado pelo conteúdo extra recebido no receptor associado a uma interface de aplicação para processamento das instâncias de objetos recebidos.

O modelo representado na Figura 14 utiliza a notação *Unified Modeling Language* (UML) (KENT, 2001), para representar três pacotes: O conteúdo principal (MC), o conteúdo extra do difusor e do receptor (BEC e REC, respectivamente). O primeiro pacote contém a abstração que representa o conteúdo típico de um programa de TV (um conjunto de fluxos de áudio e vídeo). Estas abstrações são providas por meio das classes *Vídeo* e *Áudio*. O segundo pacote contém o conjunto de classes que representa a abstração do conteúdo extra difundido, e o terceiro contém as classes que representam a abstração dos serviços de aplicação no receptor.

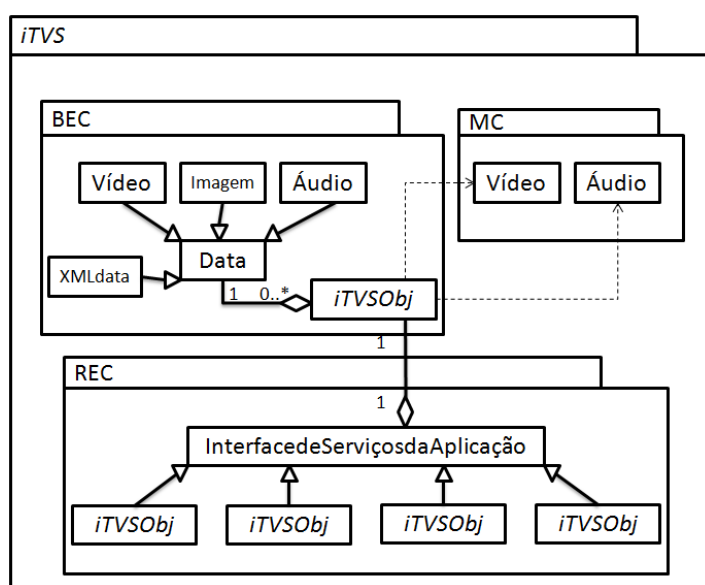


Figura 14 - Um modelo de aplicação interativa para TV em uma abordagem orientada a objetos

No pacote *BEC*, a classe *Data* é uma generalização dos dados extras difundidos. Ela pode ser especializada para representar qualquer tipo de dado extra, tais como imagens, textos, arquivos XML, vídeos, etc. A classe mais importante deste pacote é a classe *iTVSOBJ*. Esta classe representa a agregação de dados que deve ser especializada para dar suporte aos requisitos específicos da aplicação. A dependência representada entre a classe *iTVSOBJ* e as classes de conteúdo principal *Vídeo* e *Áudio*, denota que a primeira pode ser dependente da dinâmica do conteúdo principal.

No pacote *REC*, a classe mais importante é a *InterfacedeServiçosdaAplicação*, que encapsula uma lista de *iTVSOBJ*. Esta classe representa uma interface genérica de serviços de aplicação, disponível para processar instâncias de *iTVSOBJ* enviados pelo difusor. Ela pode

ser especializada para satisfazer requisitos específicos da aplicação como, por exemplo, *replays*, seleção de áudio, seleção de vídeos e visualização de dados.

A criação de objetos dinâmicos é dependente da dinâmica do programa de TV. Dentro do contexto de uma partida de futebol, por exemplo, sempre que um gol ocorre é criada uma instância do objeto que o representa, contendo como atributo o recorte do fluxo principal de áudio e vídeo onde ocorre o gol.

O conhecimento prévio da estrutura de dados extras, assim como dos diferentes tipos de eventos gerados, permite inclusive, que a aplicação faça o tratamento de evento de forma “personalizada”. Por exemplo, ao receber um evento indicando um gol, caso o evento sinalize o gol do time do coração do usuário, animações alusivas ao evento são automaticamente disparadas pela aplicação.

4.3 MODELAGEM DE UM PROGRAMA INTERATIVO DE TV AO VIVO: UMA PARTIDA DE FUTEBOL

De acordo com (MARQUES, 2009b) a estruturação dos dados que compõem a aplicação interativa é o alicerce da solução que permite o atendimento aos requisitos previamente mencionados. Com base nesta estrutura pré-definida, as mídias e seus respectivos relacionamentos podem ser adicionadas (e/ou modificadas) à aplicação interativa, em tempo de execução. Estas operações ocorrem por meio de eventos sinalizados a partir do transmissor ou mesmo de uma fonte externa.

Para ilustrar o modelo proposto foi escolhida uma aplicação interativa relacionada a um jogo de futebol com transmissão ao vivo. Para o caso específico desta aplicação, a estrutura de dados extras gerada segue o modelo ilustrado em notação orientada a objetos, representado na Figura 15.

Nesse modelo as classes *Time*, *Highlight* e *Jogador*, representam especializações da classe *iTVSObj*. Estas classes não possuem dependência em relação ao fluxo principal de áudio e vídeo. Tal característica permite que a interface de serviço de aplicação no receptor,

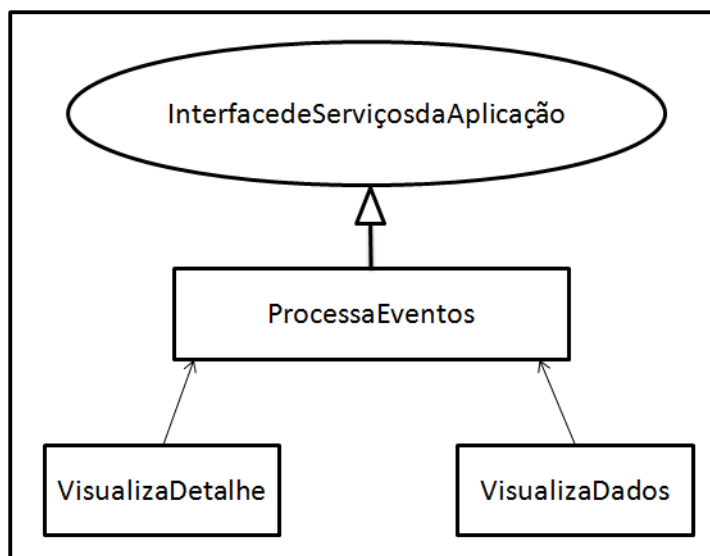


Figura 16 - Modelo de dados extras no receptor (BEC) em uma aplicação de jogo de futebol

Cabe ressaltar que os modelos ilustrados nas Figuras 15 e 16 são modelos simplificados de uma aplicação interativa para programas de TV ao vivo. A estrutura definida no modelo têm impacto direto no poder da aplicação, logo estruturas mais complexas permitem a implementação de soluções mais ricas. Neste modelo o processamento de informações, tais como a situação do jogador que comete a falta (número de cartões, histórico de faltas) e estatísticas da partida, são calculadas no próprio receptor de acordo com a implementação da aplicação.

4.4 CONSIDERAÇÕES FINAIS

A possibilidade de modificar uma aplicação interativa durante sua apresentação requer o atendimento a uma série de requisitos tanto por parte da aplicação quanto da infra-estrutura de serviços. Dentre os modelos propostos que se enquadram ao ambiente de aplicações interativas de TV, o que sugere a necessidade de estruturação dos dados extras que compõem a aplicação, e a especificação de diferentes tipos de eventos, orientados à aplicação, permite o desenvolvimento de aplicações capazes de tratar os eventos recebidos no decodificador. Estas aplicações tanto podem ser enviadas ao usuário pelo carrossel de dados, como por meio de um meio alternativo (*pendrive* ou *internet*). Uma vez que a aplicação é executada no cliente, a chegada de eventos dispara ações definidas pela própria aplicação. Tal possibilidade permite atribuir um caráter personalizado a um conteúdo de massa, de acordo com o perfil do usuário.

CAPÍTULO 5

5 AMBIENTE DE SIMULAÇÃO DE PROGRAMAS INTERATIVOS

Para validar o modelo de aplicação interativa com estruturação de dados extras e classificação de eventos, foi implementado um simulador para representar o envio de dados (áudio e vídeo principais, e extras), recepção, tratamento e apresentação de programas de TV interativos. A necessidade de implementação do simulador justifica-se devido à ausência de um ambiente procedural de TV Digital, visto que a especificação Ginga-J, embora já concluída, ainda não dispõe de uma implementação. A validação do modelo proposto em (MARQUES, 2009b) por meio da utilização de comandos de edição disponíveis para o sub-sistema declarativo Ginga-NCL foge do escopo deste trabalho.

5.1 O AMBIENTE DE SIMULAÇÃO *INTERACTIVESIM*

O ambiente de simulação *InteractiveSim* foi criado como pré-requisito para a validação de um modelo de estruturação de dados para aplicações interativas procedurais. Este modelo permite a edição dinâmica de aplicações interativas, tornando possível a atualização da aplicação, de acordo com eventos ocorridos em tempo de apresentação, no programa de TV. O tratamento a esses eventos, gerados durante a apresentação do programa, fica a cargo de uma aplicação em execução no receptor de TV. Esta aplicação, por sua vez, é modelada de acordo com o tipo de programa exibido, baseada no modelo de estrutura de dados definido para o programa. O conhecimento acerca da estrutura do modelo de dados permite que aplicações sejam desenvolvidas por terceiros (agentes fora da cadeia regular de produção de conteúdo de TV) possibilitando o desenvolvimento de aplicações de acordo com o perfil do (outrora) telespectador, agora um usuário de TV.

Vale ressaltar que o ambiente desenvolvido não implementa o conjunto de API's que constituem o Java DTV. O *InteractiveSim* é uma aplicação Java que funciona dentro dos

moldes de um sistema de transmissão de TV. Foge do escopo deste trabalho, a utilização do *InteractiveSim* na simulação e testes de programas interativos desenvolvidos com a API Java DTV.

O ambiente *InteractiveSim* é composto por quatro programas. Os detalhes de funcionamento de cada um dos seus componentes encontram-se descritos a seguir.

5.2 VISÃO GERAL DA ESTRUTURA DO AMBIENTE

O ambiente de simulação desenvolvido, o *InteractiveSim* (Figura 17), foi montado dentro dos moldes de um ambiente de TV interativa. Neste ambiente, além da emissora de TV (representada pelo *InteractiveSimSrv*), responsável pelo envio do fluxo principal de áudio e vídeo e pelos dados de programas interativos, encontra-se o receptor (representado pelo *InteractiveSimClient*), responsável pela decodificação do fluxo principal, interpretação e execução de programas interativos.

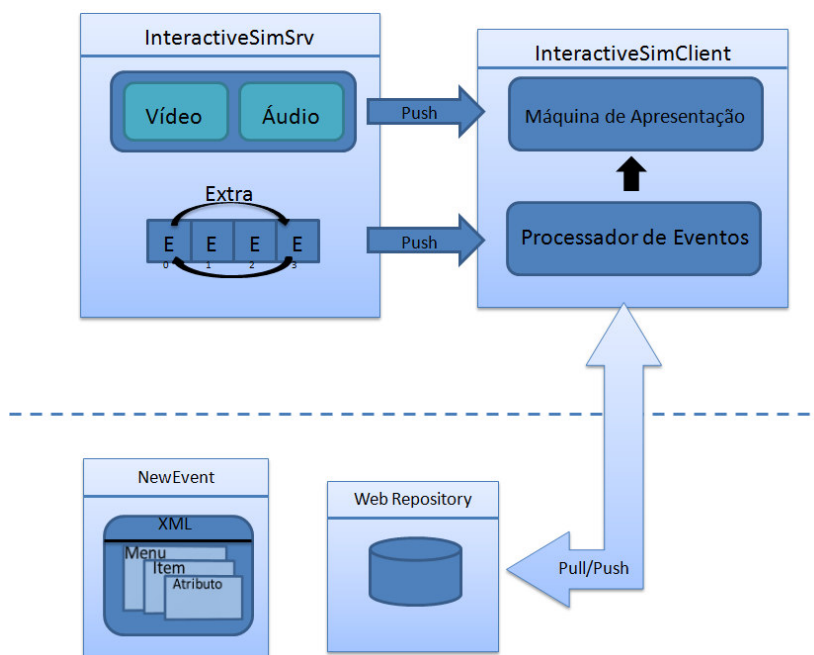


Figura 17 - Estrutura do ambiente de simulação

Outros elementos, que compõem o ambiente de simulação, representado na Figura 17, são uma fonte externa de dados de programas interativos disponível na Web e uma aplicação

responsável por definir o tipo de aplicação e a estrutura de dados que será utilizada no programa interativo.

O primeiro elemento permite o armazenamento de mídias e dados que irão compor o programa interativo, recuperados por mecanismos do tipo *pull*. Este elemento também é capaz de gerar eventos de edição na apresentação do programa, desde que o mesmo esteja previsto na estruturação dos dados da aplicação e seja suportado pela interface de serviços de aplicação no receptor. Os eventos gerados a partir desta fonte externa são transmitidos ao receptor por mecanismos do tipo *push*.

O segundo elemento funciona com um assistente para criação de modelos de dados. Ele é responsável por definir a estrutura de dados que será utilizada no programa interativo e o tipo de serviço de aplicação responsável pelo processamento desses dados. O serviço de aplicação e a estrutura de dados definida pode ser transmitida para o receptor pelo *InteractiveSimSrv*, via carrossel de dados, ou levada até o receptor por uma mídia de armazenamento externo (ex., um *pendrive*). A estrutura de dados criada pode definir a fonte das mídias a serem apresentadas no cliente. Estas fontes podem encontrar-se tanto no armazenamento local, recebidas via carrossel de dados, ou em localidades remotas, disponíveis na *Web*.

Cabe ressaltar que o ambiente desenvolvido esta focado no suporte a aplicações escritas na linguagem Java. Como elemento de interoperabilidade entre os sub-sistemas declarativo e procedural, foi definida a utilização de uma “linguagem franca” para a especificação da estrutura de dados da aplicação, o XML.

Os requisitos funcionais definidos para o ambiente foram: (i) suportar alterações dinâmicas na estrutura de apresentação, possibilitando a edição em tempo de execução da aplicação interativa; (ii) permitir que agentes externos à cadeia regular de produção de conteúdo de TV possam implementar e fornecer aplicações capazes de interagir com programas ao vivo; (iii) permitir que aplicações interativas recuperem mídias em repositórios externos (Web) e (iv) manter a coerência entre a estrutura lógica e a estrutura de apresentação em situações de sintonização tardia, problema descrito em (MARQUES, 2009b).

5.3 ARQUITETURA DO AMBIENTE

O *InteractiveSimSrv*, representado na Figura 18, é o elemento responsável pela transmissão do fluxo principal de áudio e vídeo, e pelo envio cíclico de dados que compõe o conteúdo extra do programa. A classe ativa *VideoSender* abstrai dados e funcionalidades responsáveis por gerar o fluxo principal do programa interativo e difundi-lo via RTP (Real-time Transport Protocol) (SCHULZRINNE, 2006). A classe *VideoSender* é especializada pela classe *FileSrc*, caso o fluxo principal de vídeo transmitido seja oriundo de um arquivo. Caso, o fluxo principal de vídeo seja gerado por um dispositivo de captura (*Web cam*), a especialização *CamSrc* é a responsável pela implementação da solução.

O envio cíclico de dados (carrossel de dados) é realizado via *unicast*, utilizando o protocolo de transporte TCP (TCP, 1981). Esta funcionalidade é abstraída pela classe ativa *DataSender*. Os dados extras, armazenados em um diretório pré-estabelecido, são difundidos de maneira cíclica pela instância da classe. Durante o processo de transmissão, dados de controle, permitem que a estrutura de diretórios adotada na origem seja mantida no destino. A classe *DataSender* realiza o envio de objetos representando eventos ocorridos no programa por meio da adição e atualização dos arquivos que compõem os dados de conteúdo extra.

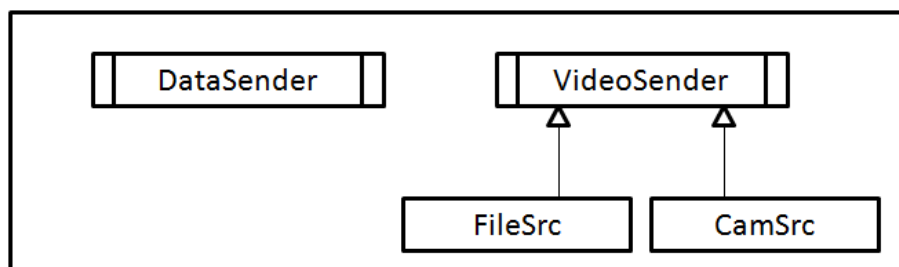


Figura 18 - Diagrama de Classes do InteractiveSimSrv

A princípio, na tentativa de simular com fidelidade o ambiente de difusão de conteúdo de uma transmissora de TV, o transporte de dados extras foi implementado em *multicast*, utilizando o protocolo de transporte UDP (POSTEL, 1980). No entanto durante testes realizados com a aplicação os seguintes problemas foram identificados:

- a) A difusão, de maneira cíclica de dados na rede, acarretava em sensível queda de desempenho na rede;

- b) A vazão do difusor era muito superior a velocidade de recepção do cliente, acarretando constantemente em arquivos recebidos incompletos;
- c) O tratamento de erros de recebimento, a cargo da aplicação, sobrecarregava o cliente, acarretando em lentidão;

Como a implementação de um mecanismo de difusão de dados, com requisitos de integridade dos arquivos recebidos, é contrária ao modelo de comunicação vigente em redes de computadores e a implementação do mesmo acarretou em diversos problemas não relacionados aos objetivos deste trabalho, tal implementação foi abandonada. Posteriormente, a implementação do mecanismo de transmissão cíclica de dados, foi desenvolvida como descrito anteriormente.

O *InteractiveSimCliente* (Figura 19) faz a decodificação e apresentação do fluxo principal de áudio e vídeo, executa aplicações e apresenta mídias. A classe *MainApp* define o *layout* da aplicação e está associada a quatro classes ativas. A classe ativa *MainPlayer* recebe o fluxo principal de vídeo enviado pelo transmissor e o reproduz. A classe *DataReceiver* armazena os dados extras enviados pelo transmissor mantendo a estrutura de diretórios utilizada na origem. A partir do primeiro ciclo completo de transmissão do carrossel de dados, ela verifica se os dados extras enviados pelo transmissor são mais recentes que as cópias locais. Caso haja arquivos novos ou mais recentes, o conteúdo transmitido no novo ciclo do carrossel sobreescreve os dados armazenados anteriormente.

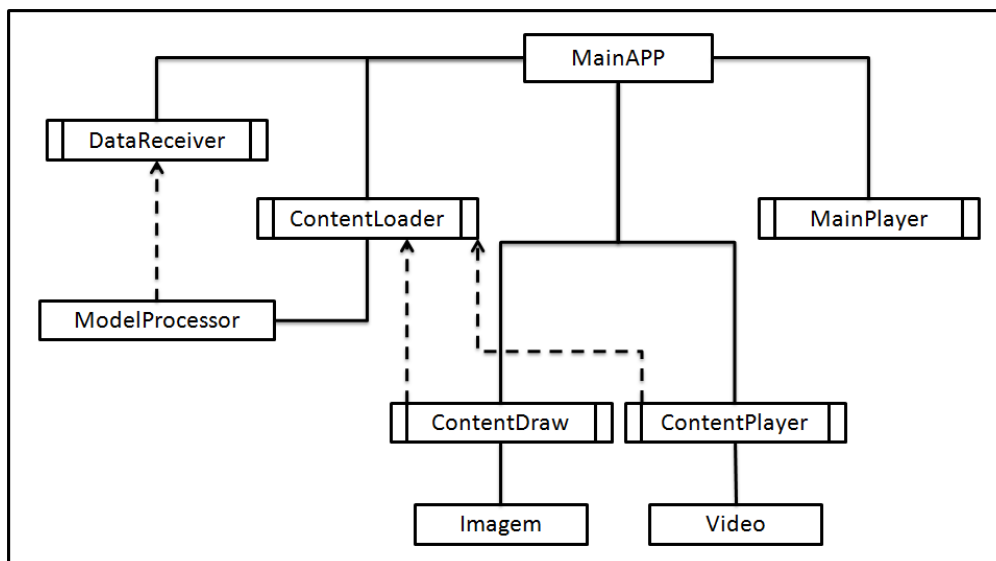


Figura 19 - Diagrama de classes do InteractiveSimClient

Uma vez que os dados extras encontram-se armazenados localmente, uma instância da classe *ModelProcessor* identifica e processa os objetos recebidos. Este processamento permite a manipulação do *layout* da aplicação. A adição de novos itens a *interface* da aplicação e a disponibilização de novos recursos interativos, fica a cargo da classe ativa *ContentLoader*. As classes *ContentDraw* e *ContentPlayer* abstraem funcionalidades associadas a exibição de vídeo e imagens extras disponíveis para a aplicação. A Figura 20, exibe a *interface* do *InteractiveSimClient*.

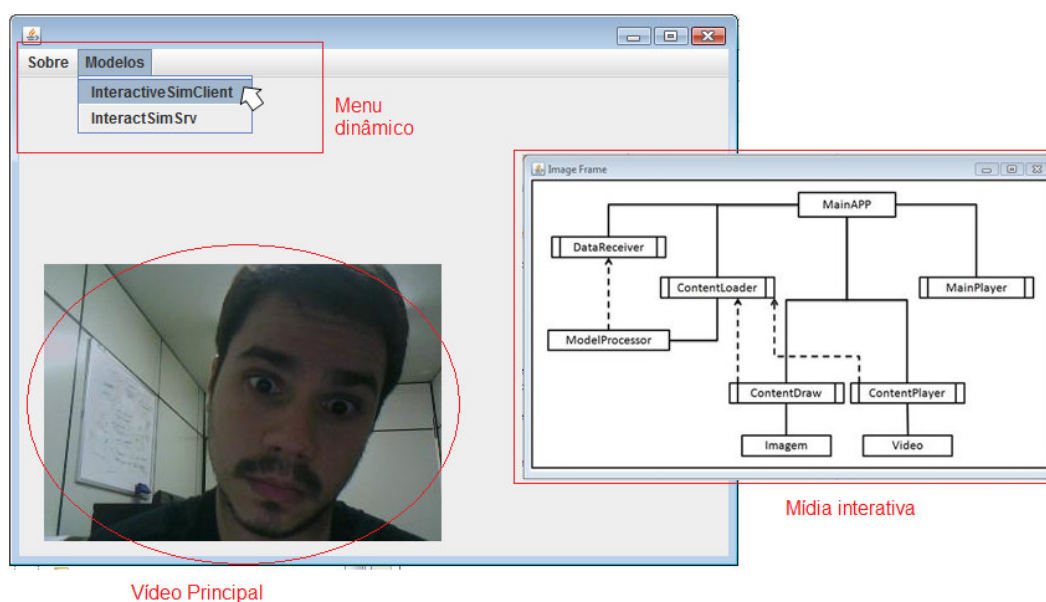


Figura 20 - *InteractiveSimClient*

No intuito de facilitar o processo de modelagem estrutural dos dados que compõem a aplicação interativa, foi desenvolvida uma ferramenta, denominada *NewEvent*. Esta ferramenta é responsável por gerar a estrutura dos dados que irão compor a aplicação de acordo com o modelo proposto em (MARQUES, 2009).

O *NewEvent* (Figura 21) é um dos agentes externos do sistema. Este componente do ambiente não possui nenhuma relação formal com o *InteractiveSimClient* e com o *InteractiveSimSrv*. A tarefa deste componente é definir a estrutura de dados que será utilizada no programa interativo e gerar arquivos que representam a ocorrência de eventos (Figura 22). Conseqüentemente também define características da aplicação. A estrutura de dados definida pode ser transmitida para o cliente pelo *InteractiveSimSrv* ou transportada em uma mídia

externa (*pendrive*). O *NewEvent* pode ainda definir, a partir dos dados estruturados, a fonte (origem) das mídias extras a serem apresentadas no cliente. Estas fontes podem encontrar-se tanto no armazenamento local, recebidas via carrossel de dados, ou em localidades remotas, disponíveis na *Web*.

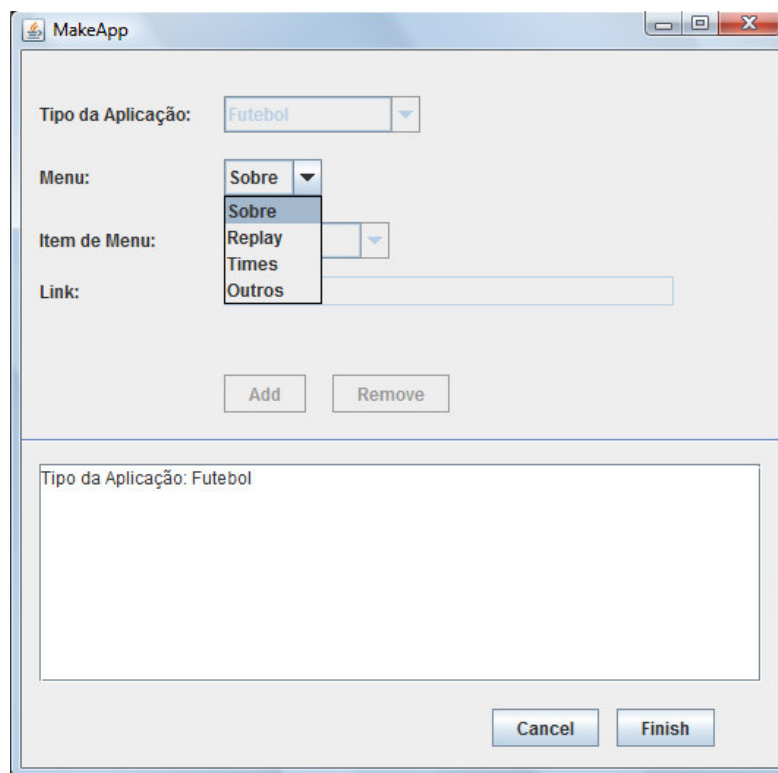


Figura 21 - Interface para definição de dados estruturados

Como em todo assistente de modelagem a capacidade de gerar modelos especializados é limitada no *NewEvent*. Para gerar a estrutura de dados de uma aplicação, o *NewEvent*, precisa que um tipo de aplicação suportada seja informada (Ex.: Futebol). Com base na seleção do tipo de aplicação desejada, é disponibilizado um conjunto de diferentes tipos de dados estruturados para adição ao modelo. Alterações entre relacionamentos estruturados, que fazem parte do tipo de aplicação modelada, não faz parte do escopo da implementação do *NewEvent*.

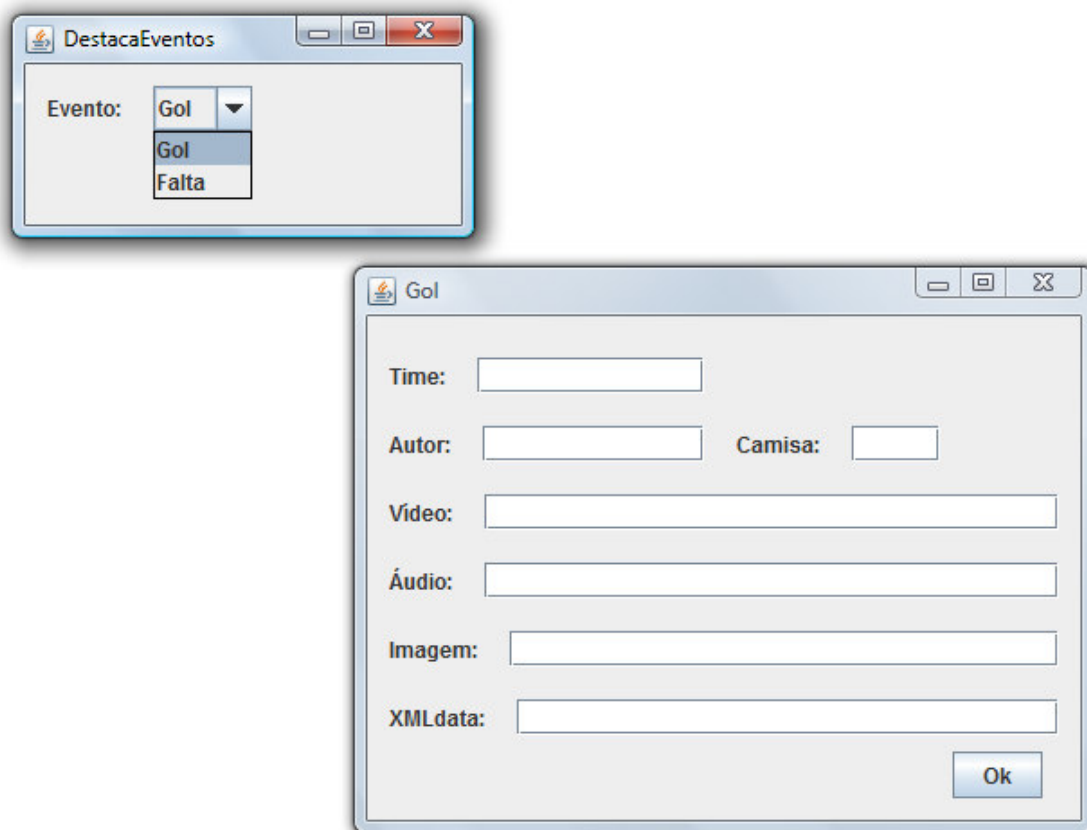


Figura 22 - Interface para geração e anotação de eventos

O *WebRepository*, como o próprio nome já diz, é um repositório de dados disponível na *Web*. Armazena dados extras, gerados no decorrer do programa interativo e também pode servir de fonte geradora de eventos de aplicação para o *InteractiveSimCliente*.

5.3.1 Dinâmica de Funcionamento

Dentre as etapas executadas no ambiente de simulação para a apresentação de programas interativos ao vivo, três fases distintas podem ser identificadas na simulação de ambiente real de produção de conteúdo interativo ao vivo, para TV.

Durante a primeira fase (Figura 23), denominada *estruturação*, é realizada a modelagem formal dos dados que compõem a estrutura do conteúdo de mídia. Nesta etapa, o processo de autoria pode ser facilitado com o uso de uma ferramenta, que a partir de *templates*

pré-estabelecidas, gera o modelo de estrutura de dados da aplicação interativa (Ex.: *NewEvent*).

Na segunda fase (Figura 23), denominada *distribuição*, o documento XML que define o modelo é adicionado ao carrossel de dados do responsável pela difusão do fluxo principal de áudio e vídeo, e dados extras (i.e.: a emissora de TV). Esse documento acompanhará os dados extras de aplicação. Outra opção para distribuição do modelo de estruturação é seu armazenamento em uma mídia alternativa, como *pendrive*, ou mesmo um servidor *Web*. Nesta última alternativa de distribuição, uma aplicação em execução no receptor faria a recuperação do documento XML, via mecanismos do tipo *pull*, e sua posterior interpretação.

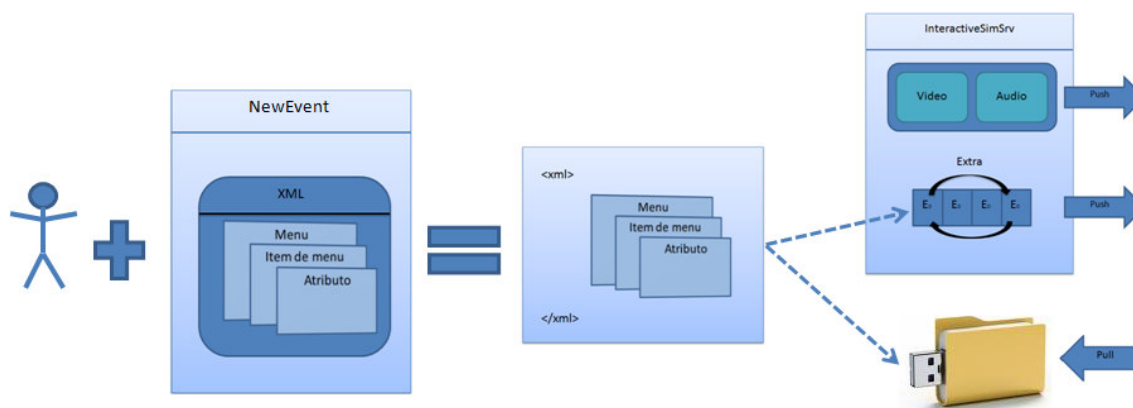


Figura 23 - Dinâmica de funcionamento para aplicações de conteúdo e formato dinâmicos - *Estruturação e Distribuição*

A seguir ocorre a fase de *operação* (Figura 24), onde eventos e dados extras são gerados e adicionados ao carrossel de dados para serem enviados ao receptor. Estas mídias, embora não descritas no documento inicial, que define a estrutura de apresentação da aplicação, são organizadas de acordo com o modelo definido na fase de *estruturação*. Portanto, embora não se tenha conhecimento prévio a respeito das mídias e relacionamentos que virão a compor a aplicação, a existência de um modelo de estruturação de dados permite que a aplicação identifique e defina em tempo de execução seus relacionamentos e restrições de acordo com tipo de dado recebido. Durante esta fase há também a possibilidade de que eventos e dados extras sejam informados ao receptor por meio de uma fonte externa disponível na *Web*. Para isso basta que a aplicação em execução no receptor estabeleça uma conexão com a fonte externa.

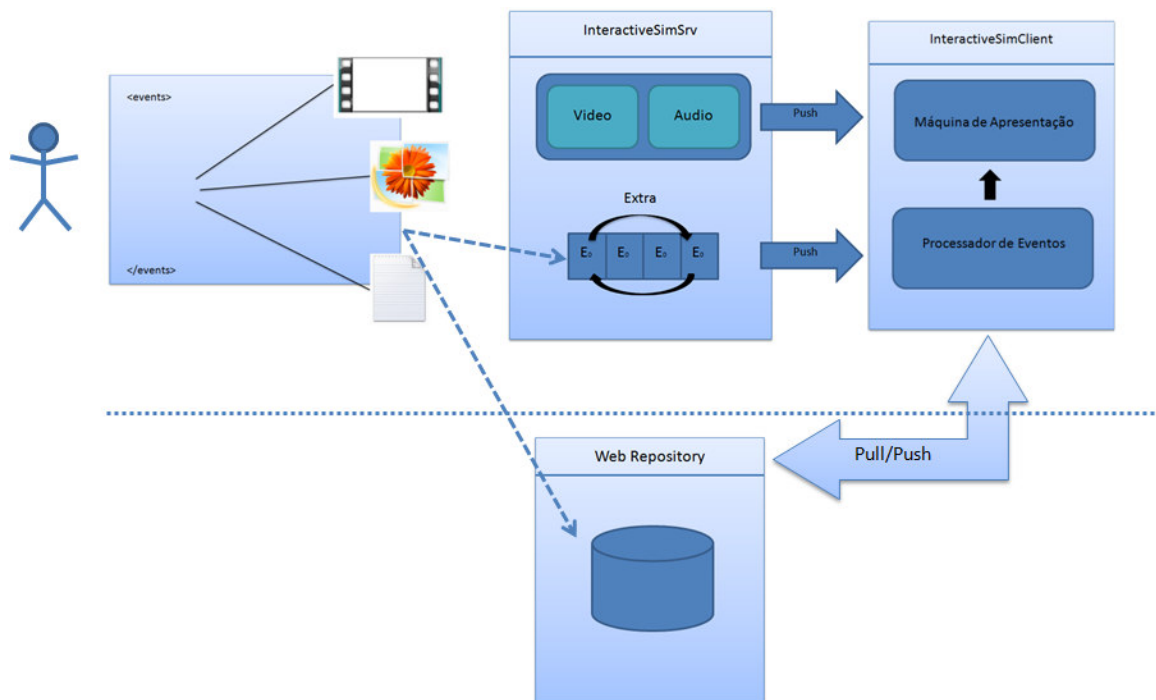


Figura 24 - Dinâmica de funcionamento para aplicações de conteúdo e formato dinâmico - *Operação*

5.4 CONSIDERAÇÕES FINAIS

O simulador implementado, visa dar suporte a validação do modelo proposto em (MARQUES, 2009b), por meio da implementação de aplicações de TVDi com relações semânticas com o conteúdo principal de áudio e vídeo, e dependência dinâmica e temporal em relação a este fluxo principal. Visto que o sub-sistema declarativo (Ginga-NCL) prevê suporte a edição dinâmica de documentos NCL, por meio de comandos de edição gerados pela transmissor, este trabalho contempla a implementação de uma modelo de edição dinâmica da aplicação interativa equivalente, em termos funcionais, ao previsto no ambiente declarativo, porem dirigido ao sub-sistema procedural (Ginga-J) do *middleware* Ginga.

CAPÍTULO 6

6 ESTUDO DE CASO

O conhecimento sobre a forma de estruturação dos dados que compõem um programa de TV interativa, e a geração, por parte do difusor, de eventos pré-estabelecidos, com relação semântica ao conteúdo principal transmitido, criam meios através dos quais se é possível confeccionar aplicações interativas com foco no perfil do usuário. Tal cenário também colabora para a disseminação e adoção de aplicações interativas, na medida em que possibilita o acesso de mais desenvolvedores, a informações-chaves para confecção de aplicações, sem que estes possuam nenhuma relação direta com o gerador/produtor do programa de TV, descentralizando o desenvolvimento de aplicações.

Este capítulo apresenta a implementação de um programa interativo de TV, com transmissão ao vivo, seguida de considerações a respeito de sua viabilidade. Convém definir *programa interativo de TV*, como um programa, onde a aplicação em execução no receptor apresenta um comportamento dinâmico ditado pelo programa de TV transmitido. Este comportamento é determinado pela ocorrência de uma série de eventos gerados pelo transmissor e esperados no receptor. Embora esperados, a ocorrência de tais eventos não pode ser prevista com exatidão.

6.1 APLICAÇÃO INTERATIVA AO VIVO – JOGO DE FUTEBOL

A transmissão de um jogo de futebol ao vivo, permite ilustrar bem o uso dos recursos oriundos da estruturação de dados e da geração de eventos, pelo transmissor. Sem dependência ao que venha a ocorrer durante a partida, podem ser enviados pelo transmissor dados contendo: a escalação dos times, o esquema tático adotado, posição atual no campeonato, etc. A aplicação interativa, por sua vez, sabendo da existência de tais dados pode tratá-los de acordo com a sua implementação. Estes dados, disponíveis desde o início da aplicação, são denominados *estáticos*. Durante o decorrer da partida, dados e eventos podem ser gerados em função da dinâmica do jogo. Gols, faltas, impedimentos, lances polêmicos,

substituições, etc., são exemplos de eventos previstos, porém sem tempo de ocorrência (ou mesmo ocorrência) bem definida durante a partida. A ocorrência de um destes eventos, pode então gerar uma ação que represente a ocorrência do evento. Este evento pode então ser comunicado a aplicação e associado a mídias e dados extras para processamento (ou exibição) no receptor. Os dados gerados em função da dinâmica da partida são então denominados, *dinâmicos*.

A classificação em dados estáticos e dinâmicos, entre os dois tipos de dados extras disponíveis para a aplicação, em execução no receptor, assim como sua organização em uma estrutura de diretórios de formato fixo, facilita o acesso aos dados, por parte da aplicação, devido ao seu conhecimento prévio a respeito da localização dos mesmos.

Na aplicação desenvolvida, a estrutura de diretórios utilizada é definida pelo transmissor. Este mesmo modelo pode ser empregado em um ambiente real, onde o radio difusor se encarregaria em definir a estrutura organizacional dos dados extras por ele enviados, aos interessados em desenvolver aplicações interativas para programas de TV.

A transmissão, e posterior acesso, aos dados *estáticos* e *dinâmicos*, é realizada por meio de sua organização na estrutura de diretórios. Essa estrutura é transmitida via carrossel de dados ao receptor e recuperada pela aplicação. A estrutura de dados dinâmicos e estáticos é representada na Figura 25.

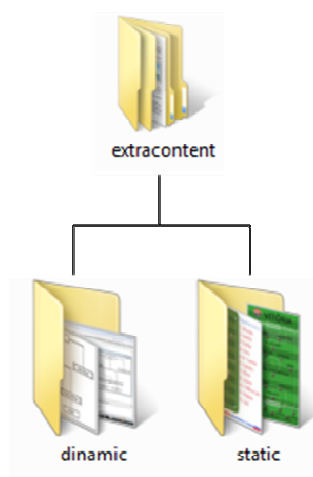


Figura 25 - Estrutura de diretórios para dados dinâmicos e estáticos

A representação da ocorrência de eventos, acontecidos durante o decorrer da partida é realizada por meio da criação de arquivos XML. Estes arquivos descrevem o evento e o associam a mídias extras (arquivos de vídeo, imagens, áudio, animações, etc.). Os arquivos XML são adicionados ao carrossel de dados, na raiz da estrutura de diretórios criada (Figura 26), e enviados ao receptor. A chegada de arquivos XML, na raiz do diretório de destino, dispara o processamento do arquivo e o tratamento das funcionalidades a ele associadas.

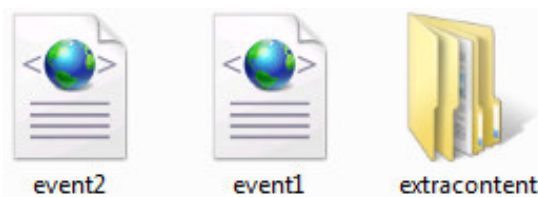


Figura 26 - Raiz de diretório: Arquivos em XML representando a ocorrência de eventos

6.1.1 Geração, Transmissão e Tratamento de Eventos

Na aplicação desenvolvida, a criação dos arquivos XML, que representam a ocorrência de eventos, é realizada pela ferramenta *NewEvent*, conforme descrito e ilustrado na Figura 22.

A ocorrência do evento, no receptor, é determinada pela chegada do arquivo XML (Figura 27). A ferramenta *NewEvent*, ao gerar o arquivo XML que representa o evento ocorrido, realiza a cópia do arquivo para o diretório raiz de origem, dos dados transportados pelo carrossel. No próximo ciclo de transporte do carrossel, o arquivo é enviado aos receptores.

```
<?xml version="1.0" encoding="ISO8859-1" ?>
  <GOL>
    <ID>1</ID>
    <AUTOR>Junior</AUTOR>
    <TEAM>Vitoria</TEAM>
    <VIDEO>test.mov</VIDEO>
  </GOL>
```

Figura 27 - Arquivo XML representando a ocorrência de um gol

No receptor, o processamento do arquivo XML, ou seja, o comportamento adotado pela aplicação, a partir da sua chegada, é dependente da implementação, o que confere ao sistema a capacidade de tratar um conteúdo genérico de forma personalizada, conforme representado nas Figuras 28 e 29.

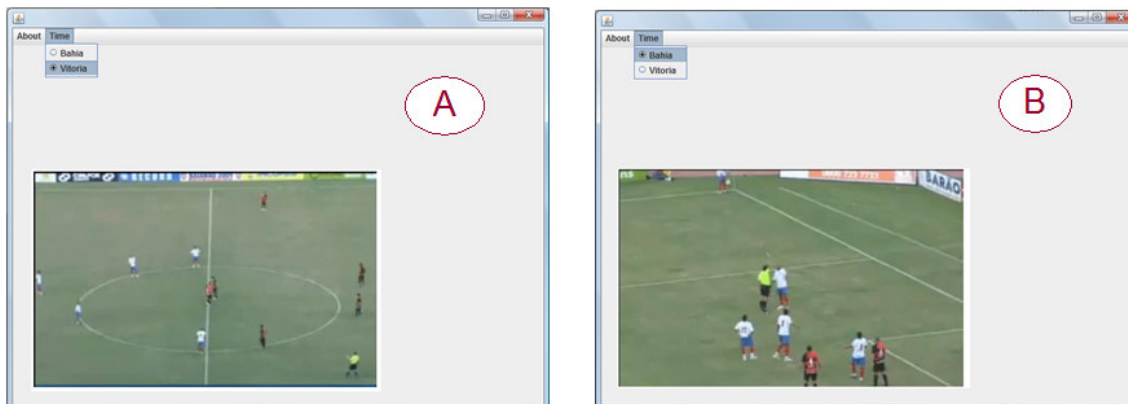


Figura 28 - Seleção, no receptor, de informação de acordo com o perfil do usuário

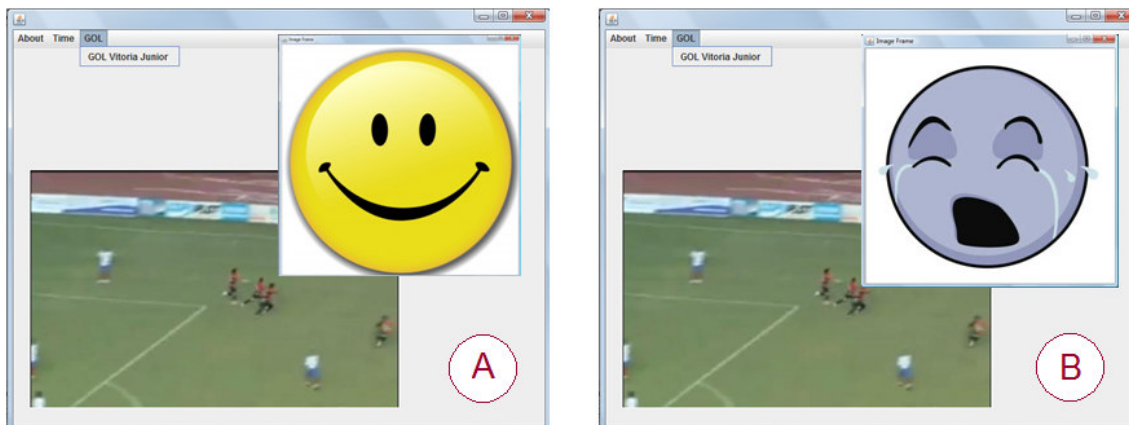


Figura 29 - Comportamento gerado de acordo com o perfil do usuário

A implementação da aplicação, no receptor, permite ao usuário a seleção de um time de sua preferência, dentre os times envolvidos na partida em questão. A montagem do *menu* que permite esta seleção pode ser realizada de acordo com arquivos de dados estáticos contendo informações sobre os times envolvidos na disputa. Uma vez selecionado o time de preferência do usuário (Figura 28), a aplicação, ao receber informações referentes a eventos, ocorridos durante o andamento da partida, pode apresentar um comportamento de acordo com o perfil do usuário (Figura 29).

No exemplo ilustrado, dois usuários (usuário A e usuário B), selecionam respectivamente Vitória e Bahia como time de sua preferência. A aplicação em execução no receptor do usuário A, ao receber a notificação do evento de GOL do time Vitória, dispara automaticamente, a apresentação de uma mídia definida de acordo com o seu perfil (Figura 29A). No receptor do usuário B, o mesmo ocorre, porém devido à sua informação de perfil, outra mídia é apresentada, após a notificação do evento de GOL (Figura 29B).

Além da apresentação automática de mídias em função da ocorrência de um determinado evento, conforme ilustrado anteriormente, a implementação cria itens de *menu*, associados ao evento ocorrido, que permitem a apresentação de mídias dinâmicas, identificadas no arquivo XML que representa o evento. Na Figura 27, o arquivo XML que representa a ocorrência de um evento do tipo GOL, associa o evento a uma mídia do tipo VIDEO identificando o arquivo de vídeo pelo nome, <VIDEO>test.mov</VIDEO> (Figura 27). A seleção do item de *menu* que identifica a ocorrência do gol, dispara a apresentação do vídeo, contendo o recorte do gol, como pode ser observado na Figura 30.

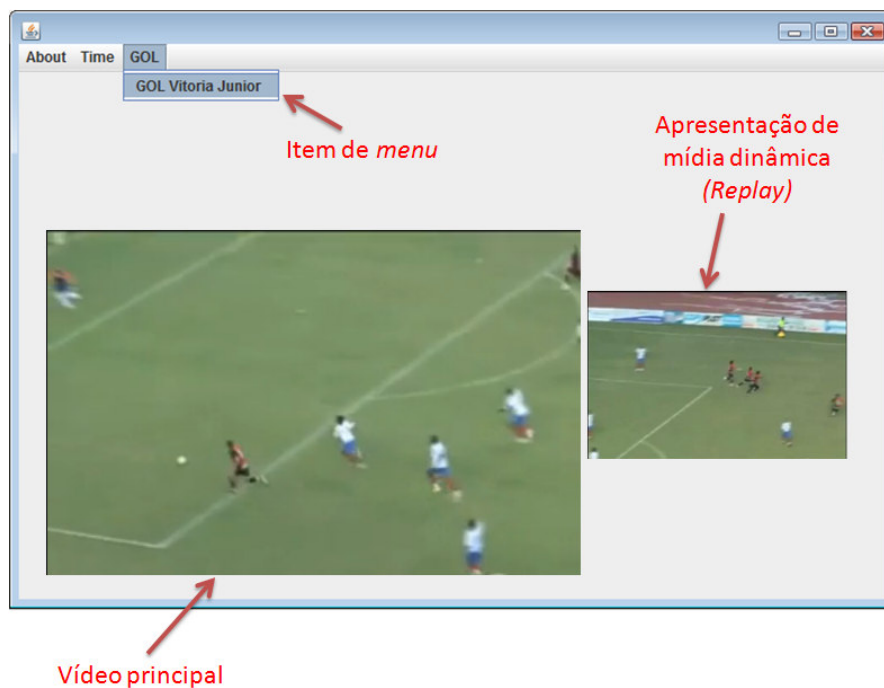


Figura 30 - Apresentação de mídias dinâmicas (extras) de acordo com solicitação do usuário

A implementação realizada contempla a ocorrência de duas classes de eventos, gerados dinamicamente pelo difusor, a classe GOL e a classe FALTA. Porém, durante uma partida de futebol, outras classes de eventos podem ser esperadas, como IMPEDIMENTO, recebimento de CARTÃO, etc. Como demonstrado, o conhecimento prévio sobre a possibilidade de ocorrência de tais eventos, acompanhados da modelagem estrutural dos elementos que o compõem, permite seu tratamento de acordo com a implementação da aplicação em execução no receptor.

6.2 CONSIDERAÇÕES

O ambiente implementado, capaz de simular o funcionamento de uma estação de TV digital, no que concerne a difusão de um fluxo de dados contendo vídeo (vídeo principal) e dados (extras), enviados de maneira cíclica pela implementação do carrossel de dados, embora fiel ao modelo de TVD interativa, utiliza redes de computadores como meio de transmissão de dados e protocolo de transporte confiável (TCP) para a entrega dos dados extras contidos no carrossel. Estas características, adicionadas ao poder computacional das máquinas utilizadas durante os testes, conferem ao ambiente de simulação um comportamento

distinto ao encontrado no ambiente real de transmissão e execução de programas de TV interativos.

Em uma transmissão de TV via radiodifusão, dados enviados pelo carrossel de dados não contam com um mecanismo de verificação e correção de erros, implementados pela camada de transporte. Ao receber um arquivo de dados, o receptor apenas confere a integridade dos mesmos, através do mecanismo de verificação de redundância cíclica (CRC) e, caso identifique a corrupção do arquivo, o receptor aguarda pelo reenvio do arquivo no próximo ciclo do carrossel, disponibilizando o início da aplicação apenas quando todos os dados que a compõem estiverem disponíveis no receptor, conforme ilustrado na Figura 31.

Esta característica pode fazer com que dados dinâmicos gerados no decorrer da apresentação demorem mais do que o esperado para que estejam disponíveis no receptor. Este fato pode levar a ocorrência de situações indesejadas, como a apresentação automática de uma animação comemorando um gol, tempos depois da ocorrência do mesmo. Outra consideração se dá ao fato da aplicação só se tornar disponível para execução, quando todos os dados que a compõem estiverem carregados na memória do receptor. Isto exige que a cada atualização do carrossel de dados a aplicação seja reiniciada no receptor, o que também pode levar a situações indesejadas, como a interrupção na apresentação de uma mídia extra, solicitada via interatividade do usuário.

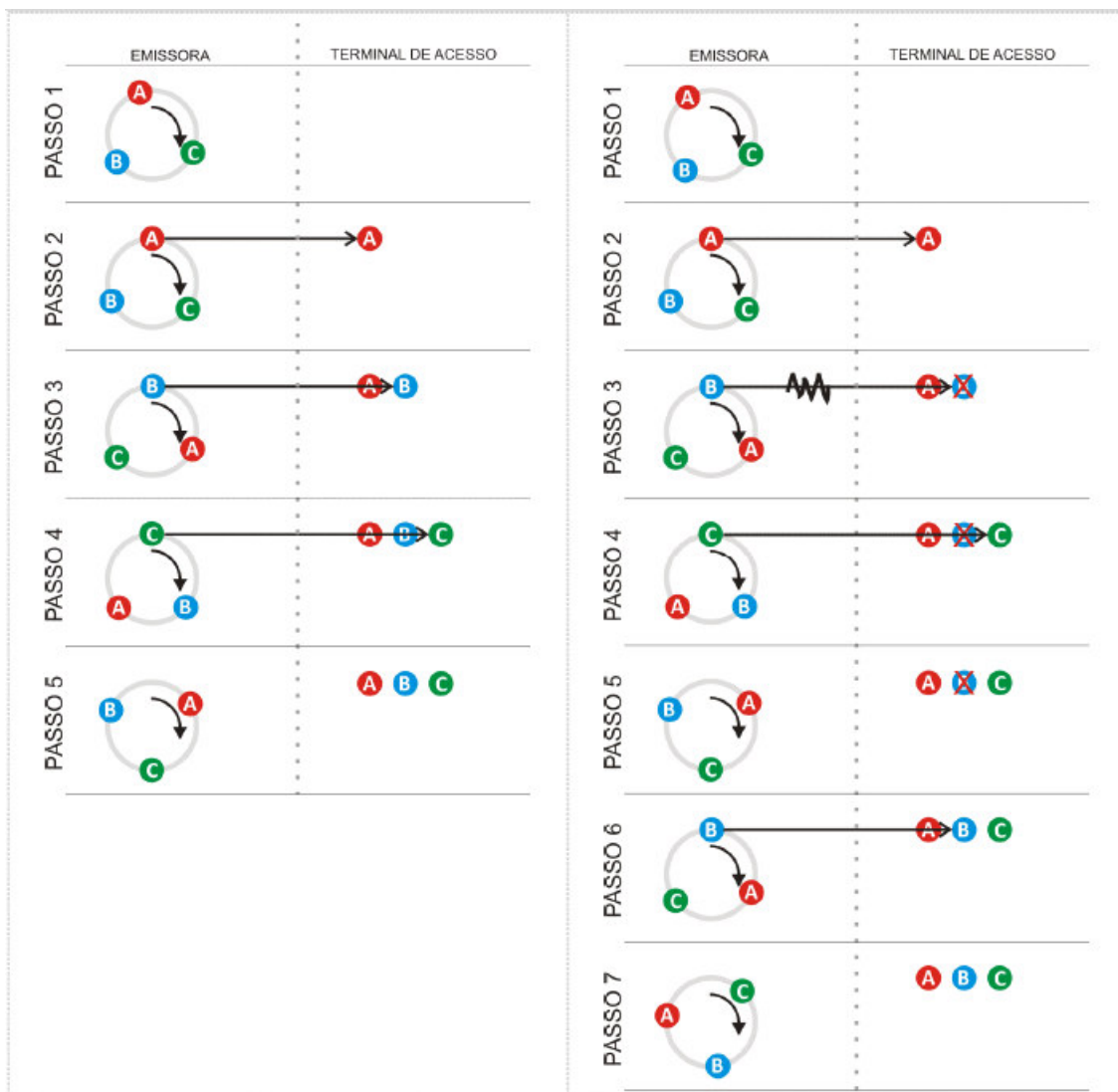


Figura 31 - Envio de dados pelo carrossel e tratamento de arquivo corrompido

Em função da maior largura de banda disponível para o envio de dados extras em uma rede de computadores, outra questão deve ser observada. Para minimizar o impacto desta característica, no que diz respeito à velocidade com que a inclusão de novos dados ao carrossel seria transmitida aos receptores, foi adicionado um atraso estimado de maneira empírica a cada novo início de um ciclo de transmissão do carrossel de dados.

No problema descrito em Marques (2009), caracterizado pela sintonização de um aparelho receptor em um programa interativo de TV, após o início do envio de dados extras, foi possível o acesso, pela aplicação, às mídias inseridas ao carrossel de dados em momentos

anteriores à sintonização. No entanto, a aplicação em execução no receptor, ao receber dados caracterizando a ocorrência de eventos passados, como gols e faltas, pode apresentar comportamento dinâmico em função dos eventos recebidos fora de seu prazo de validade, como disparar uma animação para um gol que ocorreu há vários minutos atrás. Uma possível solução para este tipo de ocorrência pode ser apresentada na forma de uma restrição temporal que indica a validade daquele evento. Caso o evento recebido não esteja dentro de seu prazo de validade, cabe à aplicação tratá-lo de maneira apropriada, como por exemplo, descartando seu conteúdo.

Para concluir, resta abordar a questão da disponibilidade de maior poder computacional no ambiente simulado. Embora o receptor possam realizar a decodificação e apresentação do fluxo principal de áudio e vídeo de maneira otimizada, em função da adoção de mecanismos de hardware (*firmwares*) com esta finalidade específica, a capacidade de execução de aplicações interativas e a manipulação de dados extras que as compõem é inferior quando comparada à plataforma PC. De acordo com o (FORUM, 2009) estarão disponíveis para o SBTVD, duas versões distintas do *middleware*. A primeira, portada para um *hardware* simplificado e sem a disponibilidade de executar aplicações interativas escritas em Java. A outra, portada em dispositivos de maior capacidade, com suporte a aplicações interativas escritas em Java. Ainda assim, em dispositivos com maior poder de processamento, a capacidade de sua memória principal pode impedir, em alguns aparelhos, a execução de algumas aplicações interativas.

Em conclusão, embora tecnicamente viável, a adoção em larga escala de tal modelo de programas interativos de TV, ainda requer o amadurecimento de questões, referentes ao ambiente de transmissão, execução e apresentação de aplicações interativas.

CAPÍTULO 7

7 CONCLUSÕES

A digitalização da transmissão dos sinais de televisão no Brasil traz uma série de benefícios aos usuários do sistema de TV. Dentre estes benefícios está a capacidade de executar, nos dispositivos receptores, aplicações típicas de computador, com as quais o usuário pode interagir.

Dentro deste contexto, essa dissertação apresentou um ambiente capaz de prover meios, que permite não só simular a interação com uma aplicação interativa executada no receptor, como também, a integração de conteúdos extras durante a transmissão ao vivo de um programa de TV.

Uma das motivações do trabalho aqui apresentado é mostrar a necessidade e provar a viabilidade da estruturação de dados extras, aliada à definição de classes de eventos que caracterizem a dinâmica do programa de TV apresentado e que são utilizados para a construção da aplicação a ser executada no receptor. A chegada, no receptor, de dados que caracterizem a ocorrência destes eventos possibilita a aplicação, o tratamento de eventos de acordo com informações de perfil do usuário. O acesso ao modelo estrutural de dados extras, disponíveis para a aplicação estimula o desenvolvimento de aplicações interativas para programas de TV, contribuindo para disseminação da tecnologia e proporcionando tratamento personalizado a um conteúdo tipicamente voltado ao consumo de massa.

Durante as discussões iniciais que definiram os rumos desta dissertação, algumas questões, levantadas durante a fase de fundamentação teórica, foram preponderantes para o desenvolvimento deste trabalho. Entre essas questões, foi possível resumir uma série de soluções a partir de algumas perguntas:

- a) Existem mecanismos nativos, que possibilitem o desenvolvimento de aplicações procedurais com relação semântica com o programa de TV apresentado, de forma que seja possível alterar o comportamento da aplicação em função da ocorrência de determinados eventos ocorridos no programa de TV? Ou seja, é possível implementar aplicações interativas dinâmicas?

Embora a parte declarativa do Ginga preveja e suporte, a edição de programas NCL, através de comandos de edição, não foi identificado nenhum mecanismo semelhante com relação ao sub-sistema procedural.

Esta constatação impeliu a busca por soluções voltadas ao sub-sistema procedural.

- b) Como é possível oferecer suporte ao desenvolvimento de aplicações interativas dinâmicas?

Os trabalhos desenvolvidos em Marques (2008, 2009b) propunham o tratamento a esta questão por meio de uma abordagem orientada a objetos, que consiste na estruturação dos dados extras que compõem a aplicação, acompanhados da definição de classes de eventos que representam a dinâmica do programa de TV.

Para validar a proposta, foi identificada a necessidade de implementar um ambiente de suporte a aplicações procedurais que representasse o ambiente de geração, distribuição e execução de aplicações interativas de TV. A necessidade de implementar tal ambiente de simulação é justificada pela ausência, até a data em questão, de uma implementação confiável do sub-sistema procedural do Ginga.

A escolha natural, por questões de compatibilidade com a proposta de implementação do sub-sistema procedural, foi utilizar a linguagem Java no desenvolvimento do ambiente e das aplicações interativas. Para isso utilizou-se o Protocolo em Tempo Real (RTP) e o Java Media Framework (JMF), para a transmissão e apresentação do fluxo de vídeo. O fato do JMF ser pouco flexível com relação a diversidade de formatos de arquivos suportados limita a capacidade do ambiente em reproduzir alguns formatos de arquivos. A implementação, além de

suportar a reprodução, em tempo real, de um fluxo de vídeo capturado de um dispositivo de captação de imagens (*webcam*) é possível reproduzir arquivos MOV e JPEG.

Em sua versão final a implementação do carrossel de dados, responsável pela entrega de dados extras aos receptores foi realizado com entrega confiável de dados (TCP) em *unicast*. Anteriormente, na tentativa de reproduzir, com fidelidade, o ambiente de entrega de dados, foi utilizada comunicação em *multicast* com transporte não confiável (UDP). Porém, questões envolvendo o sincronismo entre o envio e a recepção de dados, aliados a ocorrência de entrega de arquivos corrompidos adicionou ao ambiente a necessidade de tratar uma série de problemas não triviais que fugiam do foco do trabalho.

Para representar a ocorrência de eventos durante a exibição do fluxo principal de vídeo, optou-se pelo envio de arquivos XML que descrevem os eventos. A chegada destes arquivos no receptor caracteriza a ocorrência de um evento relacionado ao programa de TV. Na implementação realizada, estes arquivos são gerados no transmissor e adicionados a uma estrutura de diretório pré estabelecida, que permite ao receptor reconhecer, através da organização da estrutura, a finalidade dos arquivos que a compõem.

Uma vez implementados os mecanismo de entrega e apresentação do fluxo principal de vídeo e da entrega de dados extras deu-se início a implementação da aplicação interativa. Esta nova etapa apresentou mais uma questão endereçada ao trabalho.

- c) Como modelar uma aplicação interativa para um jogo de futebol transmitido ao vivo?

A implementação de aplicações interativas dinâmicas, com relação semântica ao fluxo principal de vídeo é completamente dependente da natureza do vídeo apresentado. Desta forma para modelar a aplicação foi necessário mapear informações e eventos, assim como seus relacionamentos, relevantes para um jogo de futebol. Desta forma foi então definido o modelo apresentado na Figura 4.2 (CAPÍTULO 4).

Por meio deste modelo e do conhecimento a respeito da organização dada aos arquivos que compunham a aplicação, foi possível implementar mecanismos capazes de criar *menus* e *itens de menus*, relacionando-os a eventos e arquivos de mídia extra vinculados ao programa apresentado. Este conhecimento ainda permitiu que a aplicação em execução no receptor, pudesse dispor de meios que identificassem informações referentes ao perfil do usuário. Em posse destas informações, foi então possível realizar o tratamento automático, de determinados eventos, em função do perfil definido.

A implementação desta solução envolveu o desenvolvimento de uma série de funcionalidades concorrentes de finalidade específica. Essas funcionalidades são as responsáveis pelo tratamento dos eventos esperados no receptor. Entre tais funcionalidades cabe citar a atualização automática de *menus* e *itens de menus*, a exibição, em paralelo a apresentação do fluxo de vídeo principal, de vídeos e imagens extras, e a apresentação automática de mídias em função da ocorrência de eventos de acordo com o perfil do usuário.

Para identificar qual, dentre as funcionalidade descritas, deve ser invocada em função da ocorrência de um evento, assim como que atributos do evento são necessários para gerar a instancia que representa sua ocorrência no receptor, foi implementado um *parser XML*, responsável por extrair dos arquivos que descrevem a ocorrência de eventos estas informações, e passá-las aos objetos apropriados.

7.1 CONTRIBUIÇÕES

O trabalho além de disponibilizar uma ferramenta que possibilita a simulação de um ambiente de transmissão de TV, por meio do uso de redes de computadores e dispositivos receptores implementados sobre a plataforma PC, também atesta o modelo de dados estruturados para a implementação de aplicações interativas. Estas aplicações caracterizam-se pela relação semântica com o programa de TV apresentado e a capacidade de interação com o mesmo.

Dentre as contribuições do trabalho pode-se destacar:

- a) Validação do modelo de estruturação de dados extras, proposto em (MARQUES, 2009).
- b) Implementação de uma aplicação interativa com relação semântica ao programa de TV apresentado de acordo com o modelo proposto.
- c) Definição de classes de eventos esperados durante a apresentação do programa de TV, o que possibilita o tratamento dos eventos de acordo com o perfil do usuário/telespectador, personalizando assim um conteúdo tipicamente de massa.
- d) Proposta e implementação do tratamento da sintonização tardia apresentada por (MARQUES, 2009).
- e) Utilização de um modelo de dados estruturados como base para o desenvolvimento de aplicações interativas, por interessados não integrantes da cadeia formal de produção de conteúdo de TV.
- f) Elaboração do material para o minicurso **Desenvolvimento de Aplicações para TV Digital Interativa com Ginga: Abordagem Procedural e Abordagem Declarativa** (VILAR, 2009), ministrado no IX Erbase, 2009.

7.2 TRABALHOS FUTUROS

Como modo de dar continuidade ao trabalho aqui iniciado, é proposto o desenvolvimento de uma API (*Application Program Interface*) para o desenvolvimento de aplicações interativas para programas de TV. Nesta *interface* de programa de aplicação prevê-se a inclusão de elementos comuns a uma série de aplicações voltadas para diferentes programas de TV, como também de elementos específicos encontrados em uma partida de futebol. O uso de tal *interface*, além de facilitar o processo de desenvolvimento de programas, permite abstrair e encapsular detalhes sobre os dados extras utilizados na aplicação, protegendo-a.

Tal abordagem tem sido realizada com êxito em alguns sistemas web, como o *Facebook*² e o *Twitter*³. Estes sistemas disponibilizam para a comunidade de desenvolvedores, APIs através das quais se é possível consumir informações de usuários

² www.facebook.com

³ www.twitter.com

contidas no sistema. Estas aplicações enriquecem o sistema adicionando-lhe funcionalidades e conseqüentemente atraído mais usuários, agregado assim, valor ao sistema.

REFERÊNCIAS

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 15601**: televisão digital terrestre – sistema de transmissão. Rio de Janeiro, 2008. 170.p. Disponível em: <http://www.dtv.org.br/download/pt-br/ABNTNBR15601_2007Vc_2008.pdf>. Acesso em: 2 ago. 2009.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 15606-2 Televisão digital terrestre**: codificação de dados e especificações de transmissão para radiodifusão digital parte 2: ginga-ncl para receptores fixos e móveis – linguagem de aplicação xml para codificação de aplicações. Rio de Janeiro, 2009. 215.p. Disponível em: <http://www.dtv.org.br/download/ptbr/ABNTNBR15606_2D2_2007Vc3_2008.pdf>. Acesso em: 3 ago. 2009.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 15606-4 Ginga-J**: ambiente de execução de aplicações procedurais. Rio de Janeiro, 2010. 176.p. Disponível em: <http://www.dtv.org.br/download/ptbr/ABNTNBR15606_4D2_2010.pdf>. Acesso em: 25 fev. 2010.

ADVANCED TELEVISION SYSTEMS COMMITTEE. **Advanced common application platform (ACAP)**. ATSC Proposed Standard, EUA ago. de 2005. Disponível em: <<http://www.atsc.org/standards/a101.php>>. Acesso em: 15 fev. 2009.

ANICIC, D. et al. An event-driven approach for logic-based complex event processing. In: IEEE INTERNATIONAL CONFERENCE ON COMPUTATIONAL SCIENCE AND ENGINEERING. 12., 2009, Washington, DC, USA. **Proceedings...** Washington: IEEE Computer Society, 2009, p.34-40.

ASSOCIATION OF RADIO INDUSTRIES AND BUSINESSES. **ARIB standard**. Apresentação da associação, funções e normas. Disponível em: <<http://www.arib.or.jp>>. Acesso em: 10 fev. 2009

ASSOCIATION OF RADIO INDUSTRIES AND BUSINESSES. **Data coding and transmission specifications for digital broadcasting**. volume 2: XML-based multimedia coding schema. STD-B24, Versão 4. Japão, fev. 2004. p.242.

ARNOLD, J. Frater ; PICKERING, M. **Digital television: technology and standards**. 2. ed. EUA: John Wiley & Sons Inc, 2007.

ADVANCED TELEVISION SYSTEMS COMMITTEE. **ATSC standard**. Apresentação do comitê, funções e normas. Disponível em: <<http://www.atsc.org/standards/>>. Acesso em: 10 jul. 2008.

BENBERNOU, S. et al. A spatio-temporal adaptation model for multimedia presentations. In: SEVENTH IEEE INTERNATIONAL SYMPOSIUM ON MULTIMEDIA. 7., 2005, Irvine, California. **Anais...** Irvine: IEEE, dez. 2005. p.12-20.

BENOIT, H. **Digital television: satellite, cable, terrestrial, iptv, mobile tv in the dvb framework**. 1 ed. EUA: Elsevier, 2008.

- BRACKMANN, C. **Usabilidade em Tv digital**. 2010. 199f. Dissertação (Mestrado) - Universidade Católica de Pelotas (UCPel), Pelotas-RS, 2010.
- BRASIL, Ministério da Cultura. **Projeto SBTVD : questões centrais para uma tomada de decisão**. 2003. Disponível em: <http://www.cultura.gov.br/upload/SBTVD_Minc_1143840740.pdf>. Acesso em: 02 fev. 2009.
- BROWN, A. Picard, R. **Digital terrestrial television in Europe**. 3. ed. EUA: Lawrence Erlbaum Inc, 2005.
- CASEY, L. Dsm-cc for word wide applications. In: WORKSHOP ON ACM SIGOPS EUROPEAN WORKSHOP. 7., 1996, USA, Nova York, NY, USA. **Proceedings...** New York: ACM, 1996. p.189-196.
- CESAR, T. et al. User-media interaction via interactive TV. **ACM multimedia tools and applications**, New York, USA, v.42, n.1, p.73-96, mar. 2009.
- COLLINS, W. **Fundamentals of digital television transmission**. 2 ed. EUA: John Wiley & Sons Inc, 2001.
- COURTIART, P. ; OLIVEIRA, R. Providing temporal consistency in a new multimedia synchronization model. ACM INTERNATIONAL CONFERENCE ON MULTIMEDIA '96. 4., 1996, USA, Boston, **Proceedings...** Boston: ACM, 1996. p. 32-45.
- DIGITAL VIDEO BROADCASTING. **DVB standard**. Apresentação do comitê, funções e normas. Disponível em: <<http://www.dvb.org/>>. Acesso em: 3 jul. 2008.
- ECLIPSE. **Eclipse IDE**. Disponível em: <<http://www.eclipse.org/>>. Acesso em: 22 fev. 2009.
- EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Multimedia Home Platform (MHP) specification 1.1.1**. Especificação Técnica ETSI TS 102 B12. 2000. Disponível em: <<http://ieeexplore.ieee.org/iel5/5/33232/01566632.pdf>>. Acesso em: 3 maio. 2009.
- FORUM DO SISTEMA BRASILEIRO DE TV DIGITAL. Disponível em: <<http://www.forumsbtvd.org.br/>>. Acesso em: 22 fev. 2010.
- GAWLINSKI, M. **Interactive television production**. 2. ed. EUA: Focal Press, Oxford, 2003.
- HALASZ, F.; Schwartz, M. The Dexter hypertext reference model. **Communications of the ACM**, New York, NY, USA, v.37, n.2, p.34-55, fev. 1994.
- IERUSALIMSKY, R. FIGUEIREDO, L. Celes, W. **Lua 5.0 reference manual**. Technical Report MCC -14/03. Rio de Janeiro: PUC-Rio, maio 2003.
- INTEGRATED SERVICES DIGITAL BROADCASTING. **ISBD standard**. Apresentação do comitê, funções e normas. Disponível em: <<http://www.arib.or.jp/english/>>. Acesso em: 10 de jul. 2008.
- INTERNET ENGINEERING TASK FORCE. **Transmission Control Protocol. RFC: 793**. University of Southern California, Los Angelis, CA, set. 1981. Disponível em: <<http://www.ietf.org>>. Acesso em: 12 jun. 2009.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. **Pesquisa nacional por amostra de domicílios**: síntese de indicadores 2008. Relatório Técnico. Rio de Janeiro, RJ, Brasil, 2009.

KENT, S. **Formal methods for distributed processing**: a survey of object-oriented approaches. The unified modeling language. 3. ed.. USA: Ed. Howard Bowman, 2001.

LEITE, L. et al. FlexTV: uma proposta de arquitetura de middleware para o sistema brasileiro de tv digital. **Revista de Engenharia de Computação e Sistemas Digitais**. São Paulo, n. 2, p.29-51, dez. 2005.

LUA. **Linguagem de programação Lua**. Disponível em: <<http://www.lua.org>>. 2010. Acesso em: 1 fev. 2010.

MARQUES, M. ; SANTOS, C. An event-based model for interactive live tv shows. In: PROCEEDINGS OF THE 16TH ACM INTERNATIONAL CONFERENCE ON MULTIMEDIA. 8., 2008, New York. **Proceedings...** New York, NY, USA: ACM, 2008. p.845-848.

MARQUES, M. Santos, C. StoryToCode: um modelo baseado em componentes para especificação de aplicações de tv digital e interativa convergente. SIMPÓSIO BRASILEIRO DE SISTEMAS MULTIMÍDIA E WEB. 15., 2009, Fortaleza. **Anais...** Fortaleza, CE: Webmídia, 2009a. p.46-52.

MARQUES, M. ; SANTOS, C. An approach based on events for threatening the late tuning problem in interactive live tv shows. In: EIMM '09 PROCEEDINGS OF THE 1ST ACM INTERNATIONAL WORKSHOP ON EVENTS IN MULTIMEDIA. 1., 2009b, New York, NY. **Proceedings...** USA: ACM, 2009b, p.41-48.

MENDES, L. SBTVD: uma visão sobre a tv digital no Brasil. **T&C Amazônia**. Manaus, AM. Ano 5, n. 12. p.102-120, out. 2007.

MONTEZ, C. Becker, V. TV Digital interativa: conceitos e tecnologias. SIMPÓSIO BRASILEIRO DE SISTEMAS MULTIMÍDIA E WEB. 10., 2004, Ribeirão Preto, SP. **Anais...** São Paulo: Webmídia, 2004, p.45-78.

MONTEZ, C. Becker, V. **TV digital interativa**: conceitos, desafios e perspectivas para o Brasil. 2. ed. Santa Catarina: Editora da UFSC, 2005.

MORENO, F. et al. Live editing of hypermedia documents. In: ACM SYMPOSIUM ON DOCUMENT ENGINEERING. 2006, New York, NY. **Proceedings...** USA: ACM, 2006, p.165-172.

MULTIMEDIA HOME PLATFORM. **MHP standard**. Disponível em: <<http://www.mhp.org>>. Acesso em: 7 de fev. 2010.

NESTED CONTEXT LANGUAGE. **NCL standard**. Disponível em: <<http://www.telemidia.puc-rio.br/~rmc/ncl/>>. Acesso em: 22 jan. 2010.

JAPAN BROADCASTING CORPORATION. **NHK standard**. Disponível em: <<http://www.nhk.or.jp/strl/open99/de-2/shosai-e.html>>. Acesso em: 27 jan. 2010.

OLIVEIRA, D. Shabat, T. Ginga, Iwuit, javadtv and you. In: SUN TECH DAYS 2009-2010. dez. 2009, São Paulo, SP. **Anais...**São Paulo: SUN, 2009. p.123-144. Disponível em:

<http://developers.sun.com/events/techdays/presentations/2009/pdfs/TD_SP_Ginga_Oliveria_Shabat.pdf>. Acesso em: 8 fev. 2010.

PICCOLO, L. Baranauskas, M. Desafios de design para a tv digital interativa. In: SIMPÓSIO DE FATORES HUMANOS EM SISTEMAS COMPUTACIONAI, 2006, Natal, RN. **Anais...** Natal: IHC, 2006. p.52-78.

PIMENTEL, M. et al. End-user live editing of itv programmes. **International Journal of Advanced Media and Communication**. USA, v. 4, issue 1, p.13-30, dez. de 2010.

POSTEL, J. User **datagram protocol. RFC: 768**. [S.l.]: Internet Engineering Task Force – IETF, 1980. Disponível em: <<http://www.ietf.org>>. Acesso em: 12 jun. 2009.

REDE NACIONAL DE PESQUISA. **GT Middleware**. Disponível em: <<http://www.rnp.br/pd/gts2004-2005/middleware.html>>. Acesso em: 25 de jan. de 2009.

SANT'ANNA, F. Cerqueira, SOARES, G. R. NCLua: objetos imperativos Lua na linguagem declarativa NCL. In: SIMPÓSIO BRASILEIRO DE SISTEMAS MULTIMÍDIA E WEB, 14., 2008, Vila Velha, ES. **Anais...** Vila Velha: Webmedia, 2008, p.56-73.

SCHULZRINNE, H. et al. Rtp: A transport protocol for real time applications. RFC. **Internet engineering task force – IETF**. Jul. de 1996. Disponível em: <<http://www.ietf.org>>. Acesso em: 12 jun. 2009.

SOCIEDADE BRASILEIRA DE ENGENHARIA DE TELEVISÃO. In: EVENTO SET DA REGIÃO SUL 2009. Porto Alegre, RS. **Anais...** Porto Alegre: SETSul, 2009. p.33-52. Disponível em: <www.set.com.br/eventos/setsul2009/palestras.htm>. Acesso em: 14 jul. 2009.

SOARES, L. Rodrigues R. **Nested context model 3.0 part 5: NCL (Nested Context Language) main profile**. 2005. 87f. Monografia (Ciência da Computação)- Departamento de Informática da PUC-Rio. Rio de Janeiro, 2005.

SOARES, L. Rodrigues, R. ; MORENO, M. Ginga-NCL: the declarative environment of the brazilian digital tv system. **Departamento de Informática da PUC-Rio**. Rio de Janeiro, 2006, 78f. Disponível em: <www.telemidia.puc-rio.br/>. Acesso em: 6 jun. 2008a.

SOARES, L. Barbosa, S. Tv digital interativa no Brasil se faz com ginga: fundamentos, padrões, autoria declarativa e usabilidade. **Atualizações em Informática**. Rio de Janeiro: Editora PUC-RJ, 2008b, p.76-123.

SOARES, L. Ambiente para desenvolvimento de aplicações declarativas para a tv digital brasileira. In: MULTIMEDIA DATABASES AND IMAGE COMMUNICATION. 2007, Laboratório TeleMídia, PUC-RJ. **Anais...** Rio de Janeiro: PUC-RJ, 2007. p.11-36. Disponível em: <<http://www.ncl.org.br/documentos/MDIC2007.pdf>>. Acesso em: 12 jun. 2008.

SOARES, L. Rodrigues, R. Produção de conteúdo declarativo para tv digital. In: SEMINÁRIO INTEGRADO DE SOFTWARE E HARDWARE. 2006, São Paulo, SP. **Anais...** Rio de Janeiro: Depto. de Informática PUC-Rio, 2006, p.41-56. Disponível em: <www.telemidia.puc-rio.br/>. Acesso em: 12 jun. 2008.

SOUZA, G.; Leite, L.; BATISTA, C. Ginga-J: The Procedural Middleware for the Brazilian Digital TV System. **Journal of the Brazilian Computer Society**, n. 4, v. 13. p.47-56. Porto Alegre, RS, 2007.

STANDARDIZING INFORMATION AND COMMUNICATION SYSTEMS. **ECMAScript language specification**. Padrão ECMA 262 2009. Disponível em: <<http://www.ecma-international.org/publications/standards/Ecma-262.htm>>. Acesso em: 21 dez. 2009.

SUN. **Sun Microsystems**. Disponível em: <www.sun.com>. Acesso em: 23 jan. 2010.

TAVARES, T. Santos ; ASSIS, T. C. A tv digital interativa como ferramenta de apoio à educação infantil. **Revista Brasileira de Informática na Educação**, Salvador, v.15, n.2, p.15-35, ago. 2007.

VILAR, A. Santos, C. Desenvolvimento de aplicações para tv digital interativa com Ginga: abordagem procedural e abordagem declarativa. ERBASE – ESCOLA REGIONAL BAHIA SERGIPE. 9., 2009, Ilhéus, BA. **Encontro...** Ilhéus: Universidade Estadual de Santa Cruz, 2009, p.210-294.

WORLD-WIDE WEB CONSORTIUM. **W3C Recommendation**: Namespaces in XML. Disponível em: <<http://www.w3c.org>>. Acesso em: 12 jun. 2009.